Introduction

The Kernel Toolkit is a robust set of tools developed to aid the Decentralized Hospital Computer Program (DHCP) development community and Information Resources Management (IRM) in writing, testing, and analysis of code. It is a set of generic tools that are used by developers, documenters, verifiers, and packages to support distinct tasks.

The Kernel Toolkit provides utilities for the management and definition of development projects. Many of these utilities have been used by the San Francisco Information Systems Center (ISC) for internal management and have proven valuable. Kernel Toolkit provides many programming and system management tools, and interacts directly with the underlying MUMPS (Massachusetts General Hospital Utility Multi-Programming System) environment in many different ways.

Also included in Kernel Toolkit are the following tools provided by other ISCs, and supported by the San Francisco ISC, based on their proven utility:

Multi-Term Look-Up (MTLU):

Many medical information systems depend on the standardized encoding of diagnoses and procedures for reports, searches, and statistics. The ICD DIAGNOSIS (#80), ICD OPERATIONS/PROCEDURE (#80.1), and CPT (#81) files are among some of the more critical files. The Multi-Term Look-Up utility increases the accessibility of the information in these files by associating user supplied words or phrases with terms found in a more descriptive, free-text field.

Multi-Term Look-Up enables:

- **Local set-up of virtually any reference file.**
- **Developers to modify the behavior of the "special" look-up by defining shortcuts, keywords, or synonyms.**

Multi-Term Look-Up integrates with any package that uses a reference file which has been entered in a site's LOCAL LOOKUP file (#8984.4).

Duplicate Resolution Utilities:

The Duplicate Resolution Utilities give programmers a shell that allows their users to check their data files for duplicates and merge them if any are found. They provide the functionality of combining duplicate records based on conditions established in customized applications. There are two files involved, the DUPLICATE RECORD file (#15) and the DUPLICATE RESOLUTION file (#15.1). The Merge Shell was developed by the IHS (Indian Health Service) to support their Multi-Facility Integration project.

Readers who wish to learn more about the Kernel Toolkit should consult these related manuals:

- ***Kernel Toolkit Release Notes Version 7.3***
- ***Kernel Toolkit Installation Guide V. 7.3***
- ***Kernel Toolkit User Manual V. 7.3***
- ***Kernel Toolkit Package Security Guide V. 7.3***
- **The *MIRMO/ISC Operations Document,* "Chapter 10"**
- ***Programming Standards and Conventions (SAC)***

Orientation

This manual is intended for use in conjunction with Toolkit package. Items included in the release of the Kernel Toolkit, such as routines and files, are only briefly described for quick reference. To gain a comprehensive understanding of the internal mechanisms of the Kernel Toolkit, the user needs to read the *Kernel Toolkit User Manual Version 7.3* and follow with a query of the system software itself.

This manual uses several methods to highlight different aspects of the material. Descriptive text is presented in a proportional font. "Snapshots" of computer dialogue (and other on-line displays) are shown in a non-proportional font and enclosed within a box. Editor's comments within a dialogue are displayed in italics. Italics are also used to emphasize a particular word or phrase within a sentence. The user's responses to on-line prompts are highlighted in boldface. Boldface is also used to highlight a particular topic.

The Return key is used to terminate "reads". It is illustrated as <RET> and is included in examples only when it might be unclear to the reader that such a keystroke must be entered. The following example indicates that you should enter two question marks followed by pressing the Return key when prompted to select a menu option:

```
Select Primary Menu option: ??
```

All uppercase is reserved for the representation of MUMPS code, variable names, or the formal name of options, field and file names, and security keys (e.g., the XUPROGMODE key).

After introducing the idea of a prompt and describing how it appears within the menu system, further references to that prompt might use an abbreviated form of the prompt name. For example, the "Select Primary Menu option:" prompt may be referred to as the select prompt after the initial description.

Programmer calls that are supported for use in application packages (on the Database Integration Committee (DBIC) list) are presented with a leading bullet, or indented, and include the up-arrow (^) used when calling the routine. The following is an example:

```
EN1^XQH
```

Direct mode utilities are prefaced with the MUMPS prompt to emphasize that the call is to be used *only in direct mode*. They also include the MUMPS command used to invoke the utility. The following is an example:

```
>D ^XUP
```

Implementation and Maintenance

The *Kernel Toolkit Installation Guide V. 7.3* has detailed information regarding the installation of Toolkit. Kernel V. 7.1 must be in place before Installing Toolkit. The steps for Installing Kernel V. 7.1 are explained in the *Kernel Installation Guide.* The *Kernel Toolkit Installation Guide V. 7.3* also contains many requirements and recommendations regarding how Kernel should be configured. Be sure to read the Guide before attempting to install Toolkit.

Other areas of this manual contain recommendations for global mapping, journaling, translation, and replication.

IMPLEMENTING MULTI-TERM LOOK-UP

Central Processing Unit (CPU) capacity; 3%.

Disk Space; 20,000 bytes. However, this depends on the number of entries in the LOCAL KEYWORD (#8984.1), LOCAL SHORTCUT (#8984.2), and LOCAL SYNONYM (#8984.3) files.

The Multi-Term Look-Up utility has one parameter which may be adjusted to meet the needs of an individual site. Whenever a new file is entered through the Add Entries To Look-Up File option, an additional MUMPS cross-reference is necessary on a free-text field of the new file. This reference converts the free-text field into keywords to be used in the search. In order to utilize the full functionality of the package, the cross-reference entry on the free-text field should match the INDEX field in the LOCAL LOOKUP file (#8984.4). In the following example for the ICD DIAGNOSIS file (#80), "AIHS" is entered on the free-text field as a cross-reference. "AIHS", therefore, must match the entry made at the Local Look-up INDEX prompt in the Add Entries To Look-Up File option.

Once you are in VA FileMan, do the following:

```
Select OPTION: UTILITY FUNCTIONS
Select UTILITY OPTION: CROSS-REFERENCE A FIELD

MODIFY WHAT FILE: ICD DIAGNOSIS// ICD DIAGNOSIS
                                    (12535 entries)
Select FIELD: DESCRIPTION

CURRENT CROSS-REFERENCE IS MUMPS 'D' INDEX OF FILE
CHOOSE E (EDIT)/D (DELETE)/C (CREATE): C <RET>
WANT TO CREATE A NEW CROSS-REFERENCE FOR THIS FIELD? NO// Y <RET>  (YES)
CROSS-REFERENCE NUMBER: 2// <RET>
Select TYPE OF INDEXING: REGULAR// MUMPS
```

```
WANT CROSS-REFERENCE TO BE USED FOR LOOKUP AS WELL AS FOR SORTING? YES// N
<RET>  (NO)
SET STATEMENT: S %="^ICD9(""AIHS"",I,DA)" D S^XTLKWIC
KILL STATEMENT: S %="^ICD9(""AIHS"",I,DA)" D K^XTLKWIC
INDEX: AC// AIHS
...
DO YOU WANT TO CROSS-REFERENCE EXISTING DATA NOW? YES// Y <RET>  (YES)
...EXCUSE ME, LET ME THINK ABOUT THAT A MOMENT.................
..................................................
>D ^XUP

Setting up programmer environment
Terminal Type set to: C-VT100
Select OPTION NAME: APPLICATION UTILITIES  XTMENU      Application Utilities

          Multi-Term Lookup Main Menu ...

Select Application Utilities Option: Multi-Term Lookup Main Menu

          Multi-Term Lookup (MTLU)
          Print Utility
          Utilities for MTLU ...

Select Multi-Term Lookup Main Menu Option: Utilities for MTLU

   KL     Delete Entries From Look-up
   ST     Add Entries To Look-Up File
          Add/Modify Utility ...

Select Utilities for MTLU Option: ST <RET> Add Entries To Look-Up File
Select LOCAL LOOKUP NAME: ICD DIAGNOSIS
  ARE YOU ADDING 'ICD DIAGNOSIS' AS A NEW LOCAL LOOKUP (THE 3RD)? Y <RET>
      (YES)
   LOCAL LOOKUP NAME: ICD DIAGNOSIS//  <RET>
   LOCAL LOOKUP DISPLAY PROTOCOL:  <RET>
INDEX: AIHS
...Ok, will now setup KEYWORD and SHORTCUT file DD's
   to allow terms for 'ICD DIAGNOSIS' entries...
PREFIX: M// ?
     Answer must be a unique prefix, 1-10 characters in length
PREFIX: M// D    (NOTE:  Enter the "Variable Pointer" prefix.)
  <REMINDER> Using 'Edit File', set the lookup routine, XTLKDICL, in 'ICD
      DIAGNOSIS DD
Select LOCAL LOOKUP NAME: <RET>
```

NOTE: Using the FileMan Edit File [DIEDIT] option, enter XTLKDICL at the Look-Up Program prompt. Data should be cross-referenced when installing the cross-reference. If not, data should be re-indexed after hours since this may be CPU intensive.

IMPLEMENTING DUPLICATE RESOLUTION UTILITIES

Data Storage:

Each entry in the DUPLICATE RECORD file (#15) takes approximately 500 bytes depending on the number of tests that are used and the number of packages that are affected by the record merge.

Each entry in the DUPLICATE RESOLUTION file (#15.1) takes approximately 28K depending on the number of tests that need to be run.

Data from the VAX/Alpha Performance monitor is stored in the ^XUCM global. This global grows at a rate of approximately 80k/day/node. A task can be queued to automatically keep this global purged. Raw data occupies most of this growth rate and can be retained a shorter period (1-3 months), while the daily averages in the CM DAILY STATISTICS file (#8986.6) should be retained considerably longer. This ensures its usefulness for trend analysis and other computations.

Retention:

The data in the Duplicate Record is not meant to be purged or archived. If one chose to they could purge the verified non-duplicates but this means that when the duplicate checking utilities are run these entries are put back in the DUPLICATE RECORD file (#15) and requires somebody to verify it again.

Resource Requirements:

One terminal and one printer are required. A slave printer to the terminal would be very beneficial.

Programmer Notes:

Developers need to determine if the merging of two file entries affects their package in such a way that they need to have their own unique merge that deals with only their package's files.

The following conditions usually mean that a developer has to write their own unique merge:

1. **The patient pointer field is defined as a numeric or free text field rather than a pointer.**

2. **The developer wants their end users to complete some task prior to the merge occurring.**

3. **They have compound cross-references that include the patient pointer on another field, but the cross-reference is not triggered by the changing of the patient pointer.**

4. **The Merge (Duplicate Resolution Utilities) does not do what the package developer desires.**

The following is a description of what occurs during the Merge:

The base file (e.g., PATIENT file, #2) is checked to see if it exists. Then the PT nodes (e.g., ^DD(2,0,"PT",) are checked and any false positives are removed. It then creates a list of files and fields within those files that point to the file being merged (e.g., in this example the file being merged is the PATIENT file, #2). If a file is pointing to the file being merged by its .01 field, and if that .01 field is DINUM, then all files/fields that point to that file are also gathered. The DINUM rule also applies to that file and any files pointing to it, to any depth.

Each file/field is checked and re-pointed/merged as follows:

> **If the field pointing is not a .01 field, the "from entry" is changed to the "to entry".**
>
> **If the field pointing is the .01 field but not DINUM, the "from entry" is changed to the "to entry".**
>
> **Each pointing .01 DINUM field is handled as follows:**
>
> > **If the .01 DINUM field is at the file level, ^DIT0 is called to merge the "from entry" to the "to entry" and then the "from entry" is deleted. ^DIT0 merges field by field but does not change any value in the "to entry". That means that NULL fields in the "to entry" get the value from the same field in the "from entry" if it is not NULL, and valued fields in the "to entry" remain the same. ^DIT0 also merges multiples. If a multiple entry in the "from entry" cannot be found in the "to entry", it is added to the "to entry". If a multiple entry in the "from entry" can be found in the "to entry", then that multiple entry is merged field by field.**
> >
> > **If the .01 DINUM field is at the subfile level (in a multiple), it is handled as follows:**
> >
> > > **If there is a "from entry" but no "to entry", the "from entry" is added to the "to entry", changing the .01 field value in the process, and the "from entry" is deleted.**
> > >
> > > **If there is a "from entry" and also a "to entry", the "from entry" is deleted and the "to entry" remains unchanged.**

If it is determined that a developer must have their own unique merge that deals with their files, they must make the appropriate entries in the PACKAGE file (#9.4). If they have to have some sort of action taken by end users prior to the merging of the records, they must update the MERGE PACKAGES multiple in the DUPLICATE RECORD file (#15) for that pair of records.

The following explains the entries that need to be made in the PACKAGE file (#9.4):

In your PACKAGE file (#9.4) make an entry in the AFFECTS RECORD MERGE field (#20).

In the .01 field, enter the file affected (e.g., PATIENT file, #2).

In the NAME OF MERGE ROUTINE field enter the name of your merge routine which is executed via indirection by Duplicate Resolution Utilities. If you leave this field blank but still place an entry in the PACKAGE file (#9.4), Duplicate Resolution Utilities assumes that you have some sort of interactive merge process that your end users must complete prior to the main merge occurring. It also assumes that this interactive merge process is on a separate option within the developer's package options. The values of the two records being merged are placed in:

^TMP("XDRMRGFR",$J,XDRMRG("FR"),

and

^TMP("XDRMRGTO",$J,XDRMRG("TO"),

These should be referenced by the developer if they need any certain field values since the values may have been changed prior to the execution of their merge routine.

In the RECORD HAS PACKAGE DATA field you would enter a string of MUMPS executable code that is passed the variable XDRMRG("FR") (the "from record" IEN) and set XDRZ to 0. The code should set XDRZ=1 if XDRMRG("FR") has data within your package files.

Remember to only make these entries in the PACKAGE file (#9.4) if the normal merge does not suffice for your package. If you have an entry in the PACKAGE file (#9.4) the repointing and merging as described above does not take place for those files within your Package entry.

If you leave the NAME OF MERGE ROUTINE field blank, it is assumed that you have some sort of interactive merge process that must occur prior to the main merging of the two records. At the completion of your interactive merge process the developer must set the STATUS field of the MERGE PACKAGES multiple for their package in the DUPLICATE RECORD file (#15) entry to Ready. This must be done using FileMan because of the trigger that is on the STATUS field. Once all of the MERGE PACKAGE entries have a STATUS of Ready, the main merging of the two records can occur.

CONFIGURATION FOR THE VAX/ALPHA PERFORMANCE MONITOR (VPM)

VPM requires that TaskMan be set to run with a DCL context *prior* to configuring the performance monitor's site files. To configure the CM SITE PARAMETERS (#8986.095) and CM SITE NODENAMES (#8986.3) files, run the Setup Performance Monitor option. After editing these files, the host directory and DCL command files (XUCMVPM.COM, XUCMMONITOR.COM) are created by TaskMan. An alert is sent to you once this is complete. Re-run this option whenever CPUs are added/removed from your configuration.

Using the TaskMan option Schedule/Unschedule Options [ZTMSCHEDULE] queue XUCM TASK VPM to run *hourly*. This option is the data collection driver for the VMS Monitor and checks for and loads new data into the CM DISK DRIVE RAW DATA (#8986.5) and CM NODENAME RAW DATA (#8986.51) files. Each data collection runs for 15 minutes using 10 second sample intervals (rather than the default 3 second interval). Queue the option XUCM TASK NIT to run in the early a.m., (e.g., 0001 hours). This option compiles workday averages, mails server messages, and collects "static" information such as node and hardware types. Finally, this option files selected RTHIST data and restarts RTHIST data collections for the next 24 hours.

Routine List

This chapter contains a list of the routines exported with Toolkit. Some of the renamed routines that are put in the Manager account are included. A brief description of the function or use of the routines is given.

XDRCNT **Tally records by STATUS and MERGE STATUS fields.**

XDRDADD **This routine makes the entries in the DUPLICATE RECORD file (#15).**

Called by: XDRDUP

Calls: FILE^DICN, DIE, EN^XDRMAIN

XDRDADJ **This routine is executed by a MUMPS cross-reference on the MERGE STATUS field of the DUPLICATE RECORD file (#15) only when the STATUS is set to Merged. This routine checks for entries in the file that are affected by the merging of this entry, and adjusts their .01 and .02 fields accordingly. The problem being addressed is as follows:**

1 to 5	**If 5 to 10 merged first,**	**1 to 10**
5 to 10	**then other entries would**	**5 to 10**
5 to 20	**be adjusted as follows:**	**10 to 20**

Or, if both 1 to 5 and 1 to 10 existed at the time of the merge, the 1 to 5 entry would be deleted.

The STATUS field (.03) is re-indexed because it sets cross-references based on the values in the .01 and .02 fields. Triggers are not fired for the .01, .02, or .03 fields.

Entries previously resolved are ignored.

Called by: Cross-reference on MERGE STATUS field of DUPLICATE RECORD file (#15) entry.

Calls: EN^XDRDUP, DIK

XDRDCOMP **This routine compares two file records via the Duplicate Checker algorithm.**

Calls: %ZIS, %ZISC, %ZTLOAD, DIC, DIR, EN^DITC, FILE^XDRDQUE, XDRDSCOR, XDRDUP

XDRDFPD **Find all potential duplicates for an entry in a file.**

XDRDLIST **This routine is responsible for the printing of various reports from the DUPLICATE RECORD file (#15). It prints listings of potential duplicates, ready, and not ready to merge verified duplicates.**

Calls: EN1^DIP, DIR, FILE^XDRDQUE

XDRDMAIN **This is the main driver for the duplicate checking routines.**

Calls: NOW^%DTC, DIE, DIK, XDRDPDTI, XDRDUP, XDREMSG, XDRMAINI

XDRDOC **Additional routine documentation.**

XDRDOC1 **XDRDOC continued.**

XDRDOC2 **XDRDOC continued.**

XDRDPDTI **This routine is called by XDRDMAIN when the Potential Duplicate threshold has been raised. This routine $ORDERs through the "APOT" cross-reference on the DUPLICATE RECORD file (#15), and deletes all entries that have a DC Dupe Match Score that does not meet the Potential Duplicate Threshold value. It also updates the DC POTENTIAL DUPE THRESHOLD%. It should be noted that if a person changes the weights of the Duplicate Tests, they should delete all Potential Duplicates, Unverified and rerun the Duplicate Resolution search.**

Called by: XDRDMAIN

Calls: DIE, DIK, EN^XDRDUP

XDRDPRGE **This routine enables the Duplicate Resolution manager to purge the DUPLICATE RECORD file (#15). They may purge Potential Duplicates, Verified Non-Duplicates, or both. Verified Duplicates cannot be purged until FileMan institutes some sort of archival or merged node.**

Calls: %ZTLOAD, DIC, DIR, DIK

XDRDQUE **This routine starts and stops the Duplicate Checking software when it is running in the background. If no search is running, it allows the user to queue a search to start up. If a search has been halted they may continue the search starting at the point they halted.**

Called by: XDRDCOMP, XDRDLIST, XDRDSCOR, XDRMADD

(All these calls by above are if XDRFL is undefined)
Calls: %ZTLOAD, DIC, Y^DIQ, DIR, CHECK^XDRU1, XDRCNT, XDRDFPD

XDRDSCOR **This routine sets the scores for the Duplicate Checking algorithm.**

Called by: XDRDCOMP, XDRDFPD, XDRDUP, XDRMADD, XDRMAINI

Calls: FILE^XDRDQUE, XDREMSG

XDRDSTAT **This routine displays the status of a particular search for duplicates.**

Calls: DIC, Y^DIQ

XDRDUP **This routine does the actual checking of two records and makes the determination if they are potential duplicates.**

Called by: XDRDADJ, XDRDCOMP, XDRDMAIN, XDRMADD

Calls: EN^DIQ1, XDRDADD, XDRDSCOR, XDREMSG

XDREMSG **This routine is responsible for either sending error messages to the user, or if the calling routine is running in the background, it sends a bulletin to the people in the duplicate manager mail group if one is defined.**

The meanings of XDRERR are as follows:

1 = The candidate collection routine is undefined.
2 = The candidate collection routine is not present.
3 = The potential duplicate threshold is undefined.
4 = There are no duplicate tests entered for this duplicate resolution entry.
5 = The global root node in DIC is undefined.
6 = No entry in DUPLICATE RESOLUTION file (#15.1) for this file.
7 = The From and To records are undefined.
8 = The test routine is not present.
9 = The routine defined as the pre-merge routine is not present.
10 = The routine defined as the post-merge routine is not present.
11 = The routine defined as the verified msg routine is not present.
12 = The routine defined as the merged msg routine is not present.
13 = Non-interactive merge style not allowed with DINUM files for merge entries.

Called by: XDRDMAIN, XDRDSCOR, XDRDUP, XDRMAINI, XDRU1

Calls: XMB

XDRHLP **Contains code for executable help from the DUPLICATE RECORD (#15) and DUPLICATE RESOLUTION (#15.1) files.**

XDRMADD **Adds entries to the DUPLICATE RECORD file (#15) with a status of Verified Duplicates.**

Calls: DIC, FILE^DICN, DIE, FILE^XDRDQUE, XDRDSCOR, XDRDUP, EN^XDRMAIN

XDRMAIN **Main Driver for the merge portion of the duplicate merge utilities.**

Called by: XDRDADD, XDRMADD

Calls: DIC, DIE, DIR, XDRMAINI, XDRMPACK, XDRMRG, XDRMSG, XDRMVFY

EN **Entry point for automatic merge.**

EN1 **Entry point for looping through verified ready to merge duplicates.**

EN2 **Entry point to select verified ready to merge duplicate pair.**

EN3 **Entry point to select unverified potential duplicate pair.**

XDRMAINI **Initialization routine for XDRMAIN and XDRDMAIN.**

Called by: XDRDMAIN, XDRMAIN

Calls: DIC, XDRDSCOR, XDREMSG

XDRMPACK **This routine is responsible for checking PACKAGE file (#9.4) for unique package merges and for checking these package's files to see if they have data for the merged "from" record.**

Called by: XDRMAIN

Calls: DIE

XDRMRG **This is the routine that does the actual merging of the duplicate records.**

Called by: XDRMAIN

Calls: DIE, DIK, EN^DIT0, DITM2, EN^DITMGMRG, LOCK^XDRU1

XDRMRG1 **This routine is the error trap for XDRMRG.**

Calls: %ET, DIE

XDRMSG This routine is responsible for the sending of the verified and merged messages.

Called by: XDRMAIN

Calls: XMB

XDRMVFY This routine is responsible for verifying potential duplicates.

Called by: XDRMAIN

Calls: DIE, DIR, EN^DITC

XDRPREI This is a pre-init routine for the XDR package that deletes the DUPLICATE RECORD (#15) and DUPLICATE RESOLUTION (#15.1) files' dictionaries.

XDRU This routine is a utility routine for the merge software; it does some testing for the merge software and provides the locking subroutines for the merge.

Called by: XDRDQUE, XDRMRG

Calls: XDREMSG

XINDEX The XIND* series of routines is the VA Cross-referencer. These routines are saved in the Manager's account as %IND* routines.

XINDX1 %INDEX continued.

XINDX2 %INDEX continued.

XINDX3 %INDEX continued.

XINDX4 %INDEX continued.

XINDX5 %INDEX continued.

XINDX6 %INDEX continued.

XINDX7 %INDEX continued.

XINDX8 %INDEX continued.

XINDX9 %INDEX continued.

XINDX10 %INDEX continued.

XINDX11	**%INDEX continued.**
XINDX51	**%INDEX continued.**
XINDX52	**%INDEX continued.**
XINDX53	**%INDEX continued.**
XPDKEY	**This routine provides a library of extrinsic MUMPS functions for security keys.**
XTBASE	**This routine is used in the [XT-NUMBER BASE CHANGER] option to calculate the base of a number and output the result.**
XTCMFILN	**Move Host file to mail message.**
XTEDTVXD	**This routine works with entries in the ALTERNATE EDITOR file (#1.2) to allow use of the VAX-VMS EDT and TPU editors.**
XTFC0	**Flow chart generator.**
XTFC1	**Flow chart generator.**
XTFCE	**This routine is used in the [XTFCE] option to display a flow chart of a routine from a given entry point.**
XTFCE1	**Display flow charts by entry points.**
XTFCR	**This routine is used in the [XTFCR] option to produce a flow chart of an entire routine.**
XTFCR1	**Display flowchart.**
XTINEND	**Post-init**
XTINOK	**Environment check of Init.**
XTKERM1	**Send a file using the Kermit protocol.**
XTKERM2	**Receive a file using the Kermit protocol.**
XTKERM3	**Kermit protocol send/receive.**
XTKERM4	**Utility parts of the Kermit protocol.**
XTKERMIT	**This routine is used in the [XT-KERMIT RECEIVE] option and in the [XT-KERMIT SEND] option to receive and send files using the Kermit protocol.**

XTLATSET	**This routine is used in the [XTLATSET] option to build VMS command files to coordinate the Kernel and VMS device tables. It reads from the Kernel's DEVICE file (#3.5) for _LTA devices and writes three VMS command files. The first file, LT_LOAD.COM, sets up printers in LATCP. The second, LT_PRT.DAT, is read by SYSPRINT.COM to set VMS parameters for printers and other devices and can optionally set up VMS spooling. The third, TSC_LOAD.COM, establishes printer parameters to be used in the DEC server's device tables.**
XTLKDICL	**This is the "special look-up" routine called from the DIC node of the file using MTLU.**
XTLKEFOP	**This routine contains the logic for editing keywords, shortcuts, and synonyms. It also contains the logic to kill and set Local Look-up Data Dictionaries.**
XTLKKSCH	**Processes user input and initiates the search.**
XTLKKWL	**These routines contain the logic that is the actual engine of the package.**
XTLKKWL1	**These routines contain the logic that is the actual engine of the package.**
XTLKKWL2	**These routines contain the logic that is the actual engine of the package.**
XTLKKWLD	**These routines contain the logic that is the actual engine of the package.**
XTLKMGR	**Procedure calls for developers using MTLU.**
XTLKPRT	**This routine prints the LOCAL KEYWORD (#8984.1), LOCAL SHORTCUT (#8984.2), and LOCAL SYNONYM (#8984.3) files.**
XTLKPST	**This is the postinit routine to set up the LOCAL LOOKUP file (#8984.4).**
XTLKTICD	**This routine is used to test the look-up option.**
XTLKTOKN	**Converts the user's input line to tokens.**
XTLKWIC	**This routine is used to set and kill the cross-references from the description fields.**
XTNTEG	**Routines containing exported checksum values. Call ^XTNT to determine what's changed since package installation.**

XTNTEG0 **XTNTEG continued.**

XTNTEG1 **XTNTEG continued.**

XTRCMP **This routine is used in the [XT-ROUTINE COMPARE] option to compare two routines of different names in the current account and list the differences.**

XTRGRPE **This routine provides editing of a group of routines with the %Z editor.**

XTRTHV **Produces a useful RTHIST summary.**

XTSPING **This routine is part of a Server option that takes any message sent to it and sends it back to the sender. This shows that servers are working at a site.**

XTSUMBLD **This routine builds an integrity checker of the form <namespace>NTEG. It gets the namespace from the PACKAGE file (#9.4). It then asks for a list of routines to include. There is an entry point CHECK that calculates the current checksum and displays it for selected routines. When the <namespace>NTEG routine runs, it loads the routines and recalculates the checksum, then compares it to its internal checksum. It reports "OK" if there is a match, or reports the current value if there is a difference. The ASCII value of the routine is determined as follows:**

1) **Any comment line with a single semicolon is presumed to be followed by comments and only the line tag is included.**

2) **Line 2 is excluded from the count.**

3) **The total ASCII value of the routine is determined by taking, excluding the exceptions noted above, and multiplying the ASCII value of each character by its position on the line being checked.**

XTVCHG **This routine is used in the [XT-VARIABLE CHANGER] option to change all occurrences of a variable to another.**

XTVGC1 **This routine is used by the [XTVG COMPARE] option to enter data on the global structure associated with a package into the XTV GLOBAL CHANGES file (#8991.2).**

XTVGC1A **XTVGC1 continued.**

XTVGC2	**This routine is used by the [XTVG COMPARE] option to generate a list of current global entries which differ from those previously entered into the XTV GLOBAL CHANGES file (#8991.2). The globals may differ by deletion, insertion, or by a change in content.**
XTVGC2A	**XTVGC2 continued.**
XTVGC2A1	**XTVGC2 continued.**
XTVNUM	**This routine is used in the [XT-VERSION NUMBER] option to create or update the version number of a set of routines.**
XTVRC1	**This routine is used by the [XTVR UPDATE] option to enter selected routine(s) into the XTV ROUTINE CHANGES file (#8991), or to determine whether any changes have occurred since the file was last updated. The most current version of the routine is maintained along with sufficient information about any changes to permit a detailed listing of changes in the routine.**
XTVRC1A	**This routine is used by the [XTVR MOST RECENT CHANGE DATE] option. It searches the XTV ROUTINE CHANGES file (#8991) for the most recent updating date on which a change was logged into the file.**
XTVRC1Z	**This routine is called automatically when .F is entered in the %Z editor, and calls XTVRC1 to log the changes in the routine.**
XTVRC2	**This routine is used by the [XTVR COMPARE] option, and generates a list of the changes to the program as they have been monitored by the [XTVR UPDATE] option. The changes are listed from most recent back through the number of change dates requested.**
XTVRCRES	**Restores a routine back to an older version.**
XUCMBR1	**Bernstein Response Time reports/graphs.**
XUCMBR2	**XUCMBR1 continued.**
XUCMBR3	**XUCMBR1 continued.**
XUCMBRTL	**Server that loads the Bernstein Response Time Log (BRTL) data.**
XUCMDSL	**For use by ISCs wishing to file VMS performance data from their sites.**

XUCMFGI	**Driver for installation of Filegrams and servers.**
XUCMFIL	**File data collected from VMS Monitor.**
XUCMGRAF	**Used by graph options for scaling, generating footers.**
XUCMNIT	**Processes raw data, generating morning report and mail server message.**
XUCMNIT1	**XUCMNIT continued.**
XUCMNIT2	**XUCMNIT continued.**
XUCMNI2A	**XUCMNIT continued.**
XUCMNIT3	**XUCMNIT continued.**
XUCMNT3A	**XUCMNIT continued.**
XUCMNIT4	**XUCMNIT continued.**
XUCMNIT5	**XUCMNIT continued.**
XUCMPA	**Performance assurance; compute reference ranges.**
XUCMPA1	**XUCMPA continued.**
XUCMPA2	**Performance assessments**
XUCMPA2A	**XUCMPA2 continued.**
XUCMPA2B	**XUCMPA2 continued.**
XUCMPOST	**Post-init allows the site to review settings, configure the VAX/Alpha Performance Monitor (VPM)**
XUCMPRE	**Pre-init used to move old VPM data to global nodes that match current file numbers.**
XUCMTM	**Assist with configuring TaskMan to run from DCL.**
XUCMTM1	**XUCMTM continued.**
XUCMVPG	**Disk Drive graphs.**
XUCMVPG1	**Performance Metric graphs.**
XUCMVPI	**Installs VPM directory and COM files.**

XUCMVPM Driver for raw data collection.

XUCMVPM1 XUCMVPM continued; files VMS Monitor data on VMS systems.

XUCMVPS More disk drive reports.

XUCMVPU Miscellaneous VPM functions and procedures.

XUCPCLCT This routine allows the user to schedule the starting and stopping times for data collection.

XUCPFRMT This routine is used to output the sorted data in Table or Graph format.

XUCPRAW This routine is invoked to sort, print, or kill raw data.

XUCS1E Called by XUCSTME to update the Routine/Global Accesses multiple (8987.32).

XUCS1R Contains the front end driver for the Routine/Global access report; sorted by Volume Group, within Volume Group by Date. It also contains the sort logic.

XUCS1RA Prints the data sorted by XUCS1R.

XUCS1RB Contains the front end driver for the Routine/Global access report; sorted by Date, within Date by Volume Group. It also contains the sort logic.

XUCS1RBA Prints the date sorted by XUCS1RB.

XUCS2E Called by XUCSTME to update the Global References multiple (8987.33).

XUCS2R Contains the front end driver for the Global References report; sorted by Volume Group, within Volume Group by Date. It also contains the sort logic.

XUCS2RA Prints the data sorted by XUCS2R.

XUCS2RB Contains the front end driver for the Global References report; sorted by Date, within Date by Volume Group. It also contains the sort logic.

XUCS2RBA Prints the data sorted by XUCS2RB.

XUCS4E Called by XUCSTME to update the Raw Statistics multiple (8987.34).

XUCS4R Contains the front end driver for the Raw Statistics report; sorted by Volume Group, within Volume Group by Date. It also contains the sort and print logic.

XUCS4RB Contains the front end driver for the Raw Statistics report; sorted by Date, within Date by Volume Group. It also contains the sort and print logic.

XUCS5E Not used in the current version. For the future.

XUCS5EA Not used in the current version. For the future.

XUCS6E Called by XUCSTME to update the System Configuration Parameters (field #2, SYSTEM CONFIG PARMS).

XUCS6R Prints the System Configuration Parameters.

XUCS8E Called by XUCSTME to update the Response Time multiple (8987.36).

XUCS8R Contains the front end driver for the Response Time report; sorted by Volume Group, within Volume Group by Date. It also contains the sort and print logic.

XUCS8RB Contains the front end driver for the Response Time report; sorted by Date, within Date by Volume Group. It also contains the sort and print logic.

XUCS8RG Contains the front end driver for the Response Time graph report; sorted by Volume Group, within Volume Group by Date. It also contains the sort logic.

XUCS8RGA Prints a graph of the data sorted by XUCS8RG.

XUCSCDE Called by XUCSTME to update the CPU/Disk Utilization multiple (8987.37).

XUCSCDG Contains the front end driver for the CPU/Disk Utilization graph report; sorted by Volume Group, within Volume Group by Date. It also contains the sort logic.

XUCSCDGA Prints a graph of the data sorted by XUCSCDG.

XUCSCDR Contains the front end driver for the CPU/Disk Utilization report; sorted by Volume Group, within Volume Group by Date. It also contains the sort and print logic.

XUCSCDRB	**Contains the front end driver for the CPU/Disk Utilization report; sorted by Date, within Date by Volume Group. It also contains the sort and print logic.**
XUCSLOAD	**FOR ISC USE. This is the server routine used to file incoming performance data from 486 sites if requested by the ISC.**
XUCSPRG	**Purges, based upon a site parameter, any data in file 8987.2. It has both a manual entry point and queueable entry point.**
XUCSRV	**This routine loads performance data from 486 sites into a mail message and ships it to the Capacity Management Directorate.**
XUCSTM	**Has two queueable entry points for the AM MSM-RTHIST and the PM MSM-RTHIST. It then in turn spawns MSM-RTHIST to nodes defined in File #8987.1 via Task Manager.**
XUCSTME	**This routine is used to transfer data from each nodes where MSM-RTHIST was run to the %ZRTL("XUCS",... When all of the data has been transferred it then updates File 8987.2.**
XUCSUTL	**Common sub-routine that is used throughout the XUCS* package.**
XUCSUTL2	**Common sub-routine that is used throughout the XUCS* package.**
XUCSUTL3	**Common sub-routine that is used throughout the XUCS* package.**
XURTL	**Prints system response time hourly averages from raw data.**
XURTL1	**Prints system response time bar graph of hourly averages over a selected range of dates.**
XURTL2	**Prints system response time hourly averages for several days.**
XURTL3	**Prints VAX DSM system response time bar graph of hourly averages over a selected range of dates, including CPU and DID usage.**
XURTLC	**Copies raw Response Time (RT) data into a FileMan (FM) file. It uses a significant amount of space in the MGR account.**
XURTLK	**Kills raw RT data; saves means in FM file.**
ZINDEX	**The ZIND* series of routines is the VA Cross-referencer. These routines are saved in the Manager's account as %IND* routines.**

ZINDX1	**%INDEX continued.**
ZINDX2	**%INDEX continued.**
ZINDX3	**%INDEX continued.**
ZINDX4	**%INDEX continued.**
ZINDX5	**%INDEX continued.**
ZINDX6	**%INDEX continued.**
ZINDX8	**%INDEX continued.**
ZINDX9	**%INDEX continued.**
ZINDX10	**%INDEX continued.**
ZINDX11	**%INDEX continued.**
ZINDX51	**%INDEX continued.**
ZINDX52	**%INDEX continued.**
ZINDX53	**%INDEX continued.**
ZINDXH	**%INDEX continued.**
ZOSV2MSM	**This routine is saved in the MGR account of each node defined in File 8987.1 as %ZOSV2. It has two main parts: the first is the necessary logic to start MSM's RTHIST silently, the second part is the transfer logic used to get the RTHIST data from the ^RTHIST global for each node to the %ZRTL("XUCS",... nodes.**
ZTEDIT	**This series of routines creates the generic VA routine editor as ^%Z.**
ZTEDIT1	**%Z editor - edit single lines.**
ZTEDIT2	**%Z editor continued.**
ZTEDIT3	**%Z editor - transfer lines from one place to another.**
ZTEDIT4	**%Z editor - help messages.**
ZTGS	**Global search.**
ZTMGRSET	**Set up the Manager account for the System.**

ZTP1	**This routine is called with D ^%ZTP1 to output the first and optionally second lines of routines from the current account to an output device in alphabetical, size, or date order.**
ZTPP	**This routine is called with D ^%ZTPP to produce a compressed routine print to an output device.**
ZTRDEL	**This routine may be called with D ^%ZTRDEL to specify a range of routines to delete from the current directory.**
ZTRTHV	**This routine produces response time summary output. (VAX DSM).**

File List

This chapter lists all the Toolkit files numerically by file number, indicates their global location, and gives a description for each file.

3.091 RESPONSE TIME **Global Location: ^%ZRTL(1,**

This file (which points to the RT DATE_UCI,VOL file) contains system response time averages by date, UCI and VOL, hour of day, and routine for those routines/response times which have been selected for monitoring. Data is inserted in this file by the routine XURTLK, which condenses and then purges the raw Response Time (RT) data.

3.092 RT DATE_UCI,VOL **Global Location: ^%ZRTL(2,**

This file (which is pointed to by the RESPONSE TIME file) contains unique entries for each DATE_UCI,VOL combination, as well as hourly active job averages if active job data is available. It is created by the XURTLK routine, which condenses and purges raw Response Time (RT) data.

3.094 RT RAWDATA **Global Location: ^%ZRTL(4,**

This file exists to permit the optional storage of raw response time data in VA FileMan format. The data transfer is performed by the XURTLC routine. Running that routine can be expected to triple the size of the %ZRTL global in the MGR account. A Response Time (RT) option exists to kill the file when it is no longer required.

15 DUPLICATE RECORD **Global Location: ^VA(15,**

This file is designed to analyze and resolve duplicate record problems from various data files (e.g., PATIENT file, #2). The "from" and "to" records are identified, the match status is reported, and the user initiating the process is noted. This file is cross-referenced by status and from-record.

15.1 DUPLICATE RESOLUTION **Global Location: ^VA(15.1,**

This file is used by Toolkit to facilitate duplicate checking and merging of files that have entries in the DUPLICATE RECORD file (#15). It provides the overall control information that package developers need to identify duplicates within their files and then to merge the duplicate entries.

8980 KERMIT HOLDING **Global Location: ^DIZ(8980,**

This file provides a holding place for data being transferred by the Kermit protocol. By default the data can only be accessed by the user who created it. The Kermit Menu [XT-KERMIT MENU] options may be used to send and receive data via this file. The menu also allows the creator of the data to permit access by others. This file is cross-referenced by name, creator, and access allowed to user.

8984.1 LOCAL KEYWORD **Global Location: ^XT(8984.1,**

The look-up entry (or code) can be associated with multiple key words or key phrases. The entry is displayed if the user enters all or any part of a key phrase. See an example below:

KEYPHRASE:	LOOKUP FILE:			
SALT AND PEPPER	NAME:	JOHN	HAIR COLOR:	LIGHT BROWN
SORT OF GRAY		JACK		PRETTY GRAY
SCHNAUZER		JILL		GEORGIA CLAY
		MARY		SORT OF GRAY
		JIM		BLACK AND WHITE

HAIR COLOR has an MTLU cross-reference.

Each of the key phrases above are associated with the entry JIM. Users can enter the following combinations:

- **SALT, SALT AND PEPPER, SALT & PEPPER, PEPPER, SORT OF PEPPER, SCHNAUZER returns only JIM. Note that SORT OF PEPPER returns only JIM because the tokens SORT and PEPPER must both be true for a match. PEPPER is false for MARY.**
- **SORT, SORT OF GRAY returns MARY and JIM**
- **GRAY returns MARY, JIM, and JACK**

NOTE: Look-ups are performed in the following order:

1. **SHORTCUT <-- stops here if a match is found**
2. **SYNONYM**
3. **KEYWORD**

8984.2 LOCAL SHORTCUT **Global Location: ^XT(8984.2,**

This is a word or phrase which is used *exclusively* to find an entry. During a look-up this file is checked first. If a shortcut matches the user's entry, the corresponding entry is displayed and no other look-ups are performed.

8984.3 LOCAL SYNONYM **Global Location: ^XT(8984.3,**

Synonyms are single terms that can be associated with one or more terms in the look-up file (tokens in the MTLU cross-reference). For example, CANCER can be associated with each of the specific forms of cancer that might be found. Note that if the user enters a phrase, all terms in the phrase must be true to get a match; therefore, LUNG CANCER might significantly restrict the search.

8984.4 LOCAL LOOKUP **Global Location: ^XT(8984.4,**

Files that have been configured for multi-term look-ups must be defined here, along with the name of the file's MTLU cross-reference.

8986.095 CM SITE PARAMETERS **Global Location: ^XUCM(8986.095,**

Holds parameters for the VAX/Alpha Performance Monitor drivers as well as a basic site profile. Data collection is disabled/enabled through this file.

8986.098 CM BERNSTEIN DATA **Global Location: ^XUCM(8986.098,**

All data for this file is collected by the Bernstein Response Time Monitor at the sites. The data is pre-formatted using the VMS COM file FORMAT.COM, then mailed to the server S.XUCM SERVER and to the groups defined in the CM SITE PARAMETERS file (#8986.095).

8986.3 CM SITE NODENAMES **Global Location: ^XUCM(8986.3,**

This file contains all nodenames that are monitored. Enter the name of all nodes used to support DHCP. Information for the remaining fields is collected automatically.

8986.35 CM SITE DISKDRIVES **Global Location: ^XUCM(8986.35,**

All data for this file is collected automatically and should not be edited.

8986.4 CM METRICS **Global Location: ^XUCM(8986.4,**

This file defines the data elements and associated benchmarks that should be applied to a particular hardware type. Sites should *not* modify this file. File updates are distributed via FileMan Filegram as the need arises.

8986.5 CM DISK DRIVE RAW DATA **Global Location: ^XUCM(8986.5,**

This file contains node-specific data from the VMS Monitor utility consisting of hourly collections of IO and QUEUE LENGTH.

8986.51 CM NODENAME RAW DATA **Global Location: ^XUCM(8986.51,**

This file contains node-specific data from the VMS Monitor utility related to CPU and memory utilization.

8986.6 CM DAILY STATISTICS **Global Location: ^XUCM(8986.6,**

This file is updated each evening with the average based on the raw data from the previous "workday", 8 a.m. to 4:30 p.m. It is used for generation of summary reports and server messages. Data from this file can be retained considerably longer than the raw data files, and should be most useful for trend analysis.

8991 XTV ROUTINE CHANGES **Global Location: ^XTV(8991,**

This file is used to record the most current version of a routine and information about changes which have occurred in that routine in prior versions. Routines are checked for any changes by using the [XTVR UPDATE] option which enters any changes noted and updates the most current version. There is no need for manual entry into this file.

The [XTVR COMPARE] option is used to obtain listings of the changes recorded for the routine(s) from the most recent to earlier changes.

8991.19 XTV VERIFICATION PACKAGE **Global Location: ^XTV(8991.19,**

This file is used to indicate the file numbers for the main files, and namespaces for options, keys, etc., that are to be included as a part of a package undergoing verification. This file is used to determine the files and other entries to be included by the routines used in preparing and comparing the XTV GLOBAL CHANGES file (#8991.2).

8991.2 XTV GLOBAL CHANGES Global Location: ^XTV(8991.2,

This file is used to record the state of a given verification package in terms of Data Dictionary (DD) entries, options, keys, templates, etc., for comparison with a subsequent version of the package.

Exported Options (Menu Structure)

This chapter contains Toolkit's exported menu structure. The options with any associated synonyms and their positions on the menus are shown. Following each option is any associated locks to that option.

TOOLKIT MENU TREE ROOTS

Toolkit exports the following menu trees:

- **Programmer Options [XUPROG]**

  This menu provides tools for developers and verifiers to use in writing, testing, and analysis of code.

- **Capacity Management [XTCM MAIN]**

  This menu integrates all capacity management activities into one package at the site level.

  These tools permit the monitoring of VAX/Alpha and 486 configurations for performance, response time, and resource utilization.

- **Application Utilities [XTMENU]**

  This menu contains application programming tools which provide Duplicate Resolution Utilities and increase the accessibility of medical information.

  Duplicate Resolution Utilities provide the functionality of combining duplicate records based on conditions established in customized applications.

  Multi-Term Look-Up provides a method of enhancing the look-up capabilities of associated FileMan files by permitting the use of query-like phrases.

- **Toolkit Queuable Options [XTQUEUABLE OPTIONS]**

  This menu, which has no parent, collects together all of the parentless Toolkit options that are intended to be scheduled through the TaskMan option [ZTMSCHEDULE] and not for interactive use.

- **Toolkit Options Attached To The Kernel Systems Manager Menu [EVE]**

  Two sets of menu diagrams are exported with Toolkit. They are attached to various options on the Kernel Systems Manager Menu [EVE]. They are as follows:

  1. **The Program Integrity Checker option [XUINTEG] is found in two places:**

     a. **System Security menu [XUSPY] on [EVE]**

     b. **Routine Management Menu [XUROUTINES] under the Operations Management menu [XUSITEMGR] on [EVE].**

  2. **The options Bring in Sent Routines [XTMOVE-IN] and Move Routines across Volume Sets [XTMOVE] are found in the Routine Management Menu under the Operations Management menu on [EVE]. Both of these options are locked with the XUPROGMODE security key.**

- **Kermit File Transfer Options**

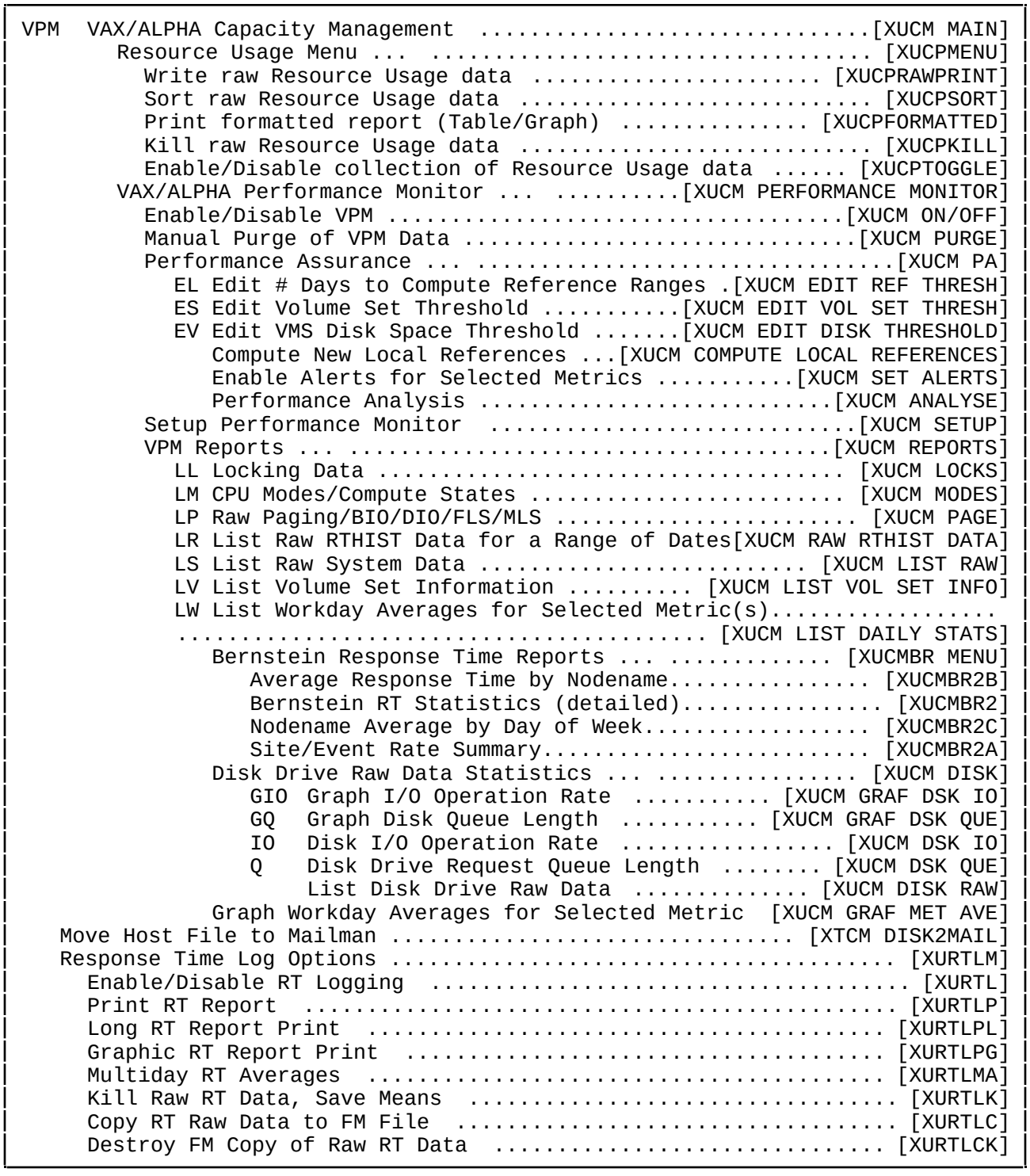
  Toolkit supports use of the Kermit file transfer protocol as an alternate editor. This allows the transfer of files from a PC or other system into a mail message or other VA FileMan word-processing field.

Programmer Options [XUPROG]

<Locked: XUPROG>

```
NTEG Build an 'NTEG' routine for a package  ..... [XTSUMBLD]
PG   Programmer mode <Locked: XUPROGMODE>  ..... [XUPROGMODE]
     Calculate and Show Checksum Values  ..... [XTSUMBLD-CHECK]
     Delete Unreferenced Options  ..... [XQ UNREF'D OPTIONS]
     Error Processing ...  ..... [XUERRS]
     Global Block Count  ..... [XU BLOCK COUNT]
     List Global <Locked: XUPROGMODE>  ..... [XUPRGL]
     Map Pointer Relations  ..... [DI DDMAP]
     Number base changer <Locked: XUPROGMODE>  ..... [XT-NUMBER BASE CHANGER]
     Routine Tools ...  ..... [XUPR-ROUTINE-TOOLS]
         %Index of Routines ..... [XUINDEX]
         Compare routines on tape to disk ..... [XUPR-RTN-TAPE-CMP]
         Compare two routines ..... [XT-ROUTINE COMPARE]
         Delete Routines <Locked: XUPROGMODE> ..... [XTRDEL]
         First Line Routine Print ..... [XU FIRST LINE PRINT]
         Flow Chart Entire Routine ..... [XTFCR]
         Flow Chart from Entry Point ..... [XTFCE]
         Group Routine Edit <Locked: XUPROGMODE> ..... [XTRGRPE]
         Input routines <Locked: XUPROG> ..... [XUROUTINE IN]
         List Routines ..... [XUPRROU]
         Output routines ..... [XUROUTINE OUT]
         Routine Edit <Locked: XUPROGMODE> ..... [XUPR RTN EDIT]
         Variable changer <Locked: XUPROGMODE> ..... [XT-VARIABLE CHANGER]
         Version number update <Locked: XUPROGMODE> ..... [XT-VERSION NUMBER]
     Test an option not in your menu <Locked: XUMGR>  ..... [XT-OPTION TEST]
     Verifier Tools Menu  ..... [XTV MENU]
         Update with current routines ..... [XTVR UPDATE]
         Routine Compare - Current with Previous ..... [XTVR COMPARE]
         Accumulate Globals for Package ..... [XTVG UPDATE]
         Edit Verification Package File ..... [XTV EDIT VERIF PACKAGE]
         Global Compare for selected package ..... [XTVG COMPARE]
         Last Routine Change Date Recorded .... [XTVR MOST RECENT CHANGE DATE]
         UNDO Edits (Restore to Older Version of Routine).....
         ..... [XTVR RESTORE PREV ROUTINE]
```

Capacity Management [XTCM MAIN]



Capacity Management [XTCM MAIN] options (continued):

(Only 486 configurations see the following menu:)

```
MSM Capacity Management Manager's Menu ..... [XUCS MANAGER MENU]
  CM Reports Menu  ..... [XUCSR REPORTS MENU]
    DATE/VG MSM CM Reports  ..... [XUCSRB REPORTS BY (DATE,VG)]
      CPU/DISK Utilization Report (By Date/VG)  .. [XUCSRB CPU/DISK REPORT]
      Global Reference Report (By Date/VG)  ..... [XUCSRB GREF REPORT]
      Response Time Report (By Date/VG)  ..... [XUCSRB RESPONSE REPORT]
      Routine CMNDS/GREF Report (by Date/VG)[XUCSRB ROU CMNDS/GREF REPORT]
      System Statistical Report (By Date/VG)  .... [XUCSRB SYS STAT REPORT]
    VG/DATE MSM CM Reports  ..... [XUCSRA REPORTS BY (VG,DATE)]
      CPU/DISK Utilization Report (By VG/Date)  .. [XUCSRA CPU/DISK REPORT]
      Global Reference Report (By VG/Date)  ..... [XUCSRA GREF REPORT]
      Response Time Report (By VG/Date)  ..... [XUCSRA RESPONSE REPORT]
      Routine CMNDS/GREF Report (by VG/Date)[XUCSRA ROU CMNDS/GREF REPORT]
      System Statistical Report (By VG/Date)  .... [XUCSRA SYS STAT REPORT]
    Graph Menu  ..... [XUCSRG GRAPHS MENU]
      Ave. %CPU & %DISK Graph  ..... [XUCSRG CPU-DISK GRAPH]
      Ave. Response Time Graph  ..... [XUCSRG RESPONSE TIME GRAPH]
  Manually Purge CM Data  ..... [XUCS MANUAL PURGE OF DATA]
  MSM Site Parameters Enter/Edit Menu  ..... [XUCS SITE EDIT MENU]
    Edit MSM CM Site Parameters  ..... [XUCS SITE EDIT]
    Enter/Edit Volume Group (Node)  ..... [XUCS VOL GROUP EDIT]
    Print/Display System Configuration Parameters  .....
      ..... [XUCS SYS CONFIG PARMS DISPLAY]
```

Application Utilities [XTMENU]

```
Duplicate Resolution System <Locked: XDR> ... ..... [XDR MAIN MENU]
   XDRO  Operations ... ..... [XDR OPERATIONS MENU]
     DSS  Display Search Status <Locked: XDR>  ... [XDR DISPLAY SEARCH STATUS]
     SRCH Start/Halt Duplicate Search  ..... [XDR SEARCH ALL]
     VPD  Verify Potential Duplicates  ..... [XDR VERIFY ALL]
     SPD  Verify Selected Potential Duplicate Pair  [XDR VERIFY SELECTED PAIR]
     MVD  Merge Ready to Merge Verified Duplicates[XDR MERGE READY DUPLICATES]
     SVD  Merge Selected Verified Duplicate Pair  .. [XDR MERGE SELECTED PAIR]
   XDRU  Utilities ... ..... [XDR UTILITIES MENU]
     CHCK Check Pair of Records to see if Duplicates  ..... [XDR CHECK PAIR]
     ADD  Add Verified Duplicate Pair  ..... [XDR ADD VERIFIED DUPS]
     FIND Find Potential Duplicates for an Entry in a File
             ..... [XDR FIND POTENTIAL DUPLICATES]
     EDIT Edit Duplicate Record Status  ..... [XDR EDIT DUP RECORD STATUS]
     VIEW View Duplicate Record Entries  ..... [XDR VIEW DUPLICATE RECORD]
     PRNT Print List of File Duplicates  ..... [XDR PRINT LIST]
     TSF  Tally STATUS and MERGE STATUS fields  .... [XDR TALLY STATUS FIELDS]
   XDRM  Manager Utilities ...  <Locked: XDRMGR> ..... [XDRMANAGER UTILITIES]
     AUTO Automatically Merge all Ready Verified Duplicates  [XDR AUTO MERGE]
     FILE Edit Duplicate Resolution File  ..... [XDR EDIT DUP RESOLUTION FILE]
     PRGE Purge Duplicate Record File  ..... [XDR PURGE]
Multi-Term Lookup Main Menu ... ..... [XTLKUSER2]
   Multi-Term Lookup (MTLU) ..... [XTLKLKUP]
   Print Utility ..... [XTLKPRTUTL]
   Utilities for MTLU ... <Locked: XTLKZMGR> ..... [XTLKUTILITIES]
     KL   Delete Entries From Look-up <Locked: XTLKZMGR>  ..... [XTLKMODPARK]
     ST   Add Entries To Look-Up File <Locked: XTLKZMGR>  ..... [XTLKMODPARS]
          Add/Modify Utility ...  ..... [XTLKMODUTL]
             SH Shortcuts  ..... [XTLKMODSH]
             KE Keywords  ..... [XTLKMODKY]
             SY Synonyms  ..... [XTLKMODSY]
```

Toolkit Queuable Options [XTQUEUABLE OPTIONS]

For Alpha Sites:

```
Compile VPM Summaries/Purge Old Data ..... [XUCM TASK NIT]
File New Data ..... [XUCM TASK VPM]
```

For 486 Sites:

```
AM MSM RTHIST Task Option..... [XUCSTASK AM RTHIST]
Tasked CM File Update..... [XUCSTASK FILE UPDATE AUTO]
PM MSM RTHIST Task Option..... [XUCSTASK PM RTHIST]
Auto Purge of CM Data..... [XUCSTASK PURGE CM DATA]
```

Toolkit Options Attached to Kernel Systems Manager Menu [EVE]

The following sets of menu diagrams are exported with Toolkit. They are attached to various options on the Kernel Systems Manager Menu [EVE] as described below.

1. **The Program Integrity Checker option [XUINTEG] reports to two separate menus on [EVE]:**

   a. **System Security menu [XUSPY] on [EVE].**

```
Systems Manager Menu  ..................................... [EVE]
   System Security ...  ................................... [XUSPY]
      Program Integrity Checker  ........................ [XUINTEG]
```

   b. **Routine Management Menu [XUROUTINES] under the Operations Management menu [XUSITEMGR] on [EVE].**

```
Systems Manager Menu  ..................................... [EVE]
   Operations Management ...  ........................... [XUSITEMGR]
      Routine Management Menu ... ..................... [XUROUTINES]
         Program Integrity Checker  ..................... [XUINTEG]
```

2. **The options Bring in Sent Routines [XTMOVE-IN] and Move Routines across Volume Sets [XTMOVE] are found in the Routine Management Menu [XUROUTINES] under the Operations Management menu [XUSITEMGR] on [EVE]. Both of these options are locked with the XUPROGMODE security key.**

```
Systems Manager Menu  ..................................... [EVE]
  Operations Management ... ............................. [XUSITEMGR]
     Routine Management Menu ... ....................... [XUROUTINES]
        Bring in Sent Routines <Locked: XUPROGMODE> ..... [XTMOVE-IN]
        Move Routines across Volume Sets <Locked: XUPROGMODE>. [XTMOVE]
```

Kermit File Transfer Options

Toolkit supports use of the Kermit file transfer protocol as an alternate editor. This allows the transfer of files from a PC or other system into a mail message or other VA FileMan word-processing field.

```
KERMIT MENU ... ..................................... [XT-KERMIT MENU]
   E  Edit KERMIT holding file ....................... [XT-KERMIT EDIT]
   R  Receive KERMIT file ........................... [XT-KERMIT RECEIVE]
   S  Send KERMIT file .............................. [XT-KERMIT SEND]
```

Cross-references

This chapter contains a description of the MUMPS-type cross-references and selected triggers that exist on fields in the Toolkit's files.

The cross-references are grouped by file. The field affected is identified along with the cross-reference's name, or number if there is no name, (X-ref ID) and a brief description.

RESPONSE TIME file (#3.091)

Field	X-ref ID	Description
DATE_UCI,VOL	"C"	This cross-reference was created to permit look-up of Response Time (RT) data by UCI,VOL.

DUPLICATE RECORD file (#15)

Field	X-ref ID	Description
RECORD1	#2	This TRIGGER sets the DATE FOUND field when an entry is added to this file.
RECORD1	#3	This TRIGGER sets the WHO CREATED field when an entry is added to this file.
RECORD2	"B"	This is a mnemonic cross-reference.
STATUS	"APOT"	The form of this cross-reference, using the PATIENT file (#2) as an example, is: ^VA(15,"APOT","DPT(",fe#1^fe#2,DA)="" The fe#s will be in the order low^high. This cross-reference will be killed when the STATUS field is changed to any other value.

DUPLICATE RECORD file (continued):

Field	X-ref ID	Description
STATUS	"ANOT"	This cross-reference will exist only when the STATUS is Verified, Not A Duplicate. The form of this cross-reference, using the PATIENT file (#2) as an example, is: ^VA(15,"ANOT","DPT(","fe#1^fe#2,DA)="" The order of the fe#s will be low^high. This cross-reference will be killed when the STATUS field is changed to any other value.
STATUS	"AVDUP"	This cross-reference permanently exists for all entries in this file that are verified duplicates. The form of this cross-reference, using the PATIENT file (#2) as an example, is: ^VA(15,"AVDUP","DPT(","fe#1^fe#2",DA)="" The order of the fe#s will be low^high.
STATUS	"ALK"	This cross-reference will exist, in one form or the other, from the time an entry is made in this file until the records are merged or verified as not a duplicate. The form of this cross-reference, using the PATIENT file (#2) as an example, is: ^VA(15,"ALK",^DPT(",fe#1,n,fe#2,DA)="" "n" will be 1 for a potential duplicate and 2 for a verified duplicate. When "n" is 1, there will be two "ALK" cross-references with the fe#s reversed. When "n" is 2, there will be only one "ALK" cross-reference and the fe#s will be in the order RECORD1 RECORD2. The "ALK2" cross-reference on MERGE DIRECTION will reset this cross-reference to be in the order "from" "to". Once merged, the "ALK" cross-reference for this entry will be killed by the "ALK3" cross-reference on the MERGE STATUS field.

DUPLICATE RECORD file (continued):

Field	X-ref ID	Description
STATUS	#5	This TRIGGER sets the DATE VERIFIED field. It is not fired for a status of Potential Duplicate, Unverified. If the status is changed from verified, the DATE VERIFIED field is deleted and will be reset if appropriate.
STATUS	#6	This TRIGGER sets the WHO VERIFIED field. It is not fired for a status of Potential Duplicate, Unverified. If the status is changed from verified, the WHO VERIFIED field is deleted and reset as appropriate. The conditional logic on this TRIGGER was modified using ^%GEDIT to prevent firing during a RE-INDEX.
STATUS	#7	This TRIGGER deletes the MERGE DIRECTION field when the status is being changed from "Verified Duplicate" to any other value. The conditional logic on this TRIGGER was modified using ^%GEDIT to prevent firing during a RE-INDEX.
STATUS	#8	This TRIGGER sets the MERGE STATUS field to 0=NOT READY when the status is set to "Verified Duplicate". The MERGE STATUS field is deleted when the status is changed from "Verified Duplicate" to any other value. The conditional logic on this TRIGGER was modified using ^%GEDIT to prevent firing during a RE-INDEX.
STATUS	#9	This TRIGGER sets the DATE RESOLVED field when the status is set to "Verified, Not A Duplicate". The DATE RESOLVED field is deleted when the status is changed to any other value.

DUPLICATE RECORD file (continued):

Field	X-ref ID	Description
STATUS	#10	This TRIGGER sets the WHO CHANGED field to the user number any time the status is changed from "Verified Duplicate" to any other value. The conditional logic on this TRIGGER was modified using ^%GEDIT to prevent firing during a RE-INDEX.
MERGE DIRECTION	"ALK2"	This cross-reference kills the existing "ALK" cross-reference for this entry and resets it to be in order "from" "to". See the "ALK" cross-reference on the STATUS field for more information.
MERGE DIRECTION	"AFR"	The "AFR" cross-reference is used by the INPUT TRANSFORMS on the .01 and .02 field to prevent entering a file entry that has already been merged away.
MERGE DIRECTION	"ATO"	The "AFR" cross-reference is used by the INPUT TRANSFORMS on the .01 and .02 field to prevent entering a file entry that has already been merged into.
MERGE STATUS	"AFR"	This cross-reference is permanent and exists for all merged entries. It indicates which file entry was the "from" entry. It is used by the INPUT TRANSFORMS on the .01 and .02 fields to prevent entering a file entry that has already been merged away. The form of this cross-reference, using File #2 as an example, is: ^VA(15,"AFR","DPT(",fe#,DA)="" fe# is the "from" file entry. Note that the kill side of this cross-reference kills all possible combinations because deleting an entry in this file using ^DIK results in the MERGE DIRECTION field being NULL when this cross-reference is fired.

DUPLICATE RECORD file (continued):

Field	X-ref ID	Description
MERGE STATUS	"ATO"	This cross-reference is permanent and exists for all merged entries. It indicates which file entry was the "to" entry. It is currently not used by the dictionary. The form of this cross-reference, using the PATIENT file (#2) as an example, is: ^VA(15,"ATO","DPT(",fe#,DA)="" fe# is the "to" file entry. Note that the kill side of this cross-reference kills all possible combinations, because deleting an entry in this file using ^DIK results in the MERGE DIRECTION field being NULL when this cross-reference is fired.
MERGE STATUS	"AMRG"	This cross-reference is short lived and exists only for entries that are verified duplicates that have not yet been merged. The form of this cross-reference, using the PATIENT file (#2) as an example, is: ^VA(15,"AMRG","DPT(",n,DA)="" "n" will be 0 for a MERGE STATUS of Not Ready and a 1 for Ready. Once merged the "AMRG" cross-reference for this entry will be killed.
MERGE STATUS	"ALK3"	This cross-reference kills the "ALK" cross-reference for this entry. See the "ALK" cross-reference on the STATUS field for more information.
MERGE STATUS	#5	This TRIGGER sets the DATE RESOLVED field when the merge status is set to "Merged". The conditional logic on this TRIGGER was modified using ^%GEDIT to prevent firing during a RE-INDEX.

DUPLICATE RECORD file (continued):

Field	X-ref ID	Description
MERGE STATUS	"ADJ"	This cross-reference is fired only when an entry has been merged. The routine ^XDRDADJ then looks at the file to determine if any other file entry pairs need to be adjusted. For example, using the PATIENT file (#2), if patient 5 was merged to patient 10, and there was a potential duplicate entry for patient 5 and patient 15, that entry would be changed to patient 10 and patient 15. There are other possible situations that are far more complex than the above example.
MFI CONTROLLED	#2	This trigger sets the MFI RESOLVED field to 0=Unresolved when a value is entered into this field. It has no effect on the change/delete side because this field is not editable.
MFI RESOLVED	"AMFIP2"	This cross-reference sets the value of the "AMFIP" cross-reference to 1 once this entry is resolved. It sets it back to "" if this field is changed from resolved.
STATUS (subfield of MERGE PACKAGES multiple)	"ANR"	This cross-reference is set only when the STATUS is "Not Ready". It is used to determine when all entries in this subfile are ready, which means the primary file entries in the DUPLICATE RECORD file (#15) entry may now be merged.
STATUS (subfield of MERGE PACKAGES multiple)	#2	This TRIGGER sets the MERGE STATUS field in this subfiles parent file. The MERGE STATUS field is set to a value computed by the computed field READY in this subfile. The value will compute to 0=Not Ready if there is any entry in this subfile that is not ready. It will compute to 1=Ready only after all entries in this subfile have said they are ready. This TRIGGER must not be modified to fire on the kill side.

DUPLICATE RESOLUTION file (#15.1)

Field	X-ref ID	Description
FILE TO BE CHECKED	"AGL"	This cross-reference is utilized by the XDRDUP when adjusting existing score values for a Duplicate Record entry.
POTENTIAL DUPLICATE THRESHOLD%	"APDTI"	This cross-reference is set whenever the Potential Duplicate Threshold is increased. This cross-reference is utilized by the Duplicate Resolution software to let it know to go through the existing Non-verified Potential Duplicates and see if the duplicate record pair meet the increased Potential Duplicate Threshold. If not, the duplicate record pair entry is deleted from the file. The variable XDR("APDTI") is left around if somebody deletes the entry from the DUPLICATE RESOLUTION file (#15.1). This is due to FileMan never allowing you to know if a person is just editing, adding, or deleting an entry.
FILE FOR INFORMATION	"AZ1"	This cross-reference is used to make FileMan log the response so that the input transform on the FIELD TO BE CHECKED can refer to the $P value of this field.

KERMIT HOLDING file (#8980)

Field	X-ref ID	Description
CREATOR	#1	Trigger cross-reference.
NAME	#2	Needed as part of the security screen.

LOCAL KEYWORD file (#8984.1)

Field	X-ref ID	Description
ENTRY	"AC"	This cross-reference passes the keyword or "tokenized" phrase into the special look-up cross-reference of the target file.
NAME	"AIHS"	This cross-reference passes the keyword or tokenized phrase into the special look-up cross-reference of the target file in the same manner as is done on the ENTRY field.

LOCAL SHORTCUT file (#8984.2)

Field	X-ref ID	Description
FREQUENTLY USED NARRATIVE	"AC"	This cross-reference is of the form: ^XT(8984.2,"AC",global root,shortcut,DA)
ENTRY	"AD"	Resets the "AC" cross-reference (normally set when editing FREQUENTLY USED NARRATIVE).

LOCAL SYNONYM file (#8984.3)

Field	X-ref ID	Description
SYNONYM	"AA"	Associates the look-up file # with the synonym and the MTLU term. Takes the form: ^XT(8984.3,"AA",LOOKUP FILE,SYNONYM,MTLU TERM)
TERM	"AC"	Associates the synonym with the global root of the look-up file.
ASSOCIATED FILE	"AD"	Associates the synonym with the global root of the look-up file in the "AC" cross-reference.

Archiving and Purging

ARCHIVING

There are no package-specific archiving procedures or recommendations for Toolkit.

For the Duplicate Resolution Utilities each merged record pair is meant to stay stored in the DUPLICATE RECORD file (#15). At some point in time, when FileMan has implemented some sort of merge node for its records, archiving could be done.

PURGING

Toolkit provides several options to facilitate the purging of Toolkit files and the cleanup of Toolkit-produced globals. The chart below contains a list of the purging options. The recommended scheduling frequency is shown for some options; all those options are queuable. The location of a detailed discussion of each option is given; unless otherwise noted, the reference given is to a chapter in the *Kernel Toolkit User Manual V. 7.3*.

Purging Option	Reference for Detailed Info.
Destroy FM Copy of Raw RT Data [XURTLCK]	Capacity Management
Kill Raw RT Data, Save Means [XURTLK]	Capacity Management
Kill Raw Resource Usage Data [XUCPKILL]	Capacity Management (Alpha Sites)
Manually Purge CM Data [XUCS MANUAL PURGE OF DATA]	Capacity Management (486 Sites)
Manual Purge of VPM Data [XUCM PURGE]	Capacity Management (Alpha Sites)

There are no purging requirements in the Multi-Term Look-Up utility.

The Duplicate Resolution Utilities provide the capability to purge all records in the DUPLICATE RECORD file (#15) that have a status of either verified non-duplicates or unverified potential duplicates. You cannot purge entries that are verified duplicates. The penalty for the purging of these records is that the duplicate checking algorithm checks to see if the records are already in the DUPLICATE RECORD file (#15) and if they are it doesn't enter them again. This saves processing time and also the user's time in re-verifying a pair as not duplicates.

Callable Routines

This chapter contains two lists of entry points into routines that are available for general use. The first list consists of calls that can be used in other applications. The second list contains utilities that can only be used directly from the MUMPS prompt. In addition, several extrinsic functions that can be used in applications or from programmer mode are mentioned.

Every entry point, extrinsic function, and executable node is described in the *Kernel Toolkit User Manual V. 7.3*. Refer to the indicated chapter in that manual for details, including input and output variables for the calls.

APPLICATION ENTRY POINTS

Entry Point	Description	Chapter
RECEIVE^XTKERMIT	Receive File Using Kermit	Tools
SEND^XTKERMIT	Send File Using Kermit	Tools
$$SY^XTLKMGR	Add synonyms to LOCAL SYNONYM file (#8984.3)	MTLU
$$K^XTLKMGR	Add keywords to LOCAL KEYWORD file (#8984.1)	MTLU
$$SH^XTLKMGR	Add shortcuts to LOCAL SHORTCUT file (#8984.2)	MTLU
$$L^XTLKMGR	Add entries to LOCAL LOOKUP file (#8984.4)	MTLU
$$DSH^XTLKMGR	Delete shortcuts	MTLU
$$DSY^XTLKMGR	Delete synonyms	MTLU
$$DK^XTLKMGR	Delete keywords	MTLU
$$DLL^XTLKMGR	Delete entry from LOCAL LOOKUP file (#8984.4)	MTLU
$$LKUP^XTLKMGR	General Lookup Utility	MTLU

DIRECT MODE UTILITIES

Entry Point	Description	Chapter
>D ^%INDEX	Check and Verify Routine	Tools
>D ^nsNTEG	Check Integrity of namespace (ns) Package	Tools
>D ONE^nsNTEG	Check Integrity Routine in namespace (ns) Package	Tools
>D ^XTBASE	Change Number Base	Tools
>D ^XTFCE	Produce Entry Point Flow Chart	Tools
>D ^XTFCR	Produce Routine Flow Chart	Tools
>D ^XTLATSET	Build VMS Startup Command File	Tools
>D ^XTRCMP	Compare Routine	Tools
>D TAPE^XTRCMP	Compare Routine (tape to disk)	Tools
>D ^XTRGRPE	Edit Group of Routines	Tools
>D ^XTSUMBLD	Create Integrity Check Routines	Tools
>D CHECK^XTSUMBLD	Calculate Checksum	Tools
>D ^XTVCHG	Change Variable	Tools
>D ^XTVNUM	Update Version Number	Tools
>X ^%Z	Edit Routine	Tools
>J ^ZTCPU	Capture Usage Data (M/SQL)	Tools
>D CDPLOT^ZTCPU	Graph Usage Report (M/SQL)	Tools
>D PRINT^ZTCPU	Print Usage Report (M/SQL)	Tools
>D PURGE^ZTCPU	Purge Usage Log (M/SQL)	Tools
>D STOP^ZTCPU	Stop Usage Data Collection (M/SQL)	Tools
>D ^%ZTP1	Print Routine First Line	Tools
>D ^%ZTPP	List Routines	Tools
>D ^%ZTRDEL	Delete Routine	Tools
>D ^ZTRTHV	Summarize Usage Reports (VAX DSM)	Tools

External Relations

TOOLKIT'S PLACE IN DHCP

Toolkit provides a set of generic tools which are used by developers, system managers, documenters, verifiers, and packages to support distinct tasks. These tools have been developed to aid the Decentralized Hospital Computer Program (DHCP) development community and Information Resources Management (IRM) in writing, testing, and analysis of code. Toolkit fully integrates with VA FileMan V. 20.0 and Kernel V. 7.1.

MULTI-TERM LOOK-UP

MTLU interacts with any DHCP package that is using a file in the LOCAL LOOKUP file (#8984.4).

DUPLICATE MERGE

These routines send bulletins to users about potential duplicates and merged records when MailMan is installed to deliver messages. The FileMan utilities to compare and transfer two entries are utilized by the XDR* routines. Any files that are going to be checked for duplicate entries must first be an entry in the DUPLICATE RECORD file (#15). The XDR* routines make calls to package developer written routines to determine if two records are potential duplicates. The XDR* routines also check the PACKAGE file (#9.4) to check for packages that are affected by that record's merge.

TOOLKIT'S EXTERNAL RELATIONS WITH THE MUMPS OPERATING SYSTEMS

Toolkit depends upon the presence of one of the American National Standards Institute (ANSI) MUMPS environments it supports. Micronetics Standard MUMPS (MSM) and VAX Digital Standard MUMPS (VAX DSM) have become the primary ANSI MUMPS environments supported by Toolkit. DataTree MUMPS (DTM), InterSystems Standard MUMPS+ for the PDP-11 (M/11+), and MSM-Unix are also supported. Low priority support of the VAX M/SQL is also still maintained.

Operating system interfaces are involved in each aspect of Toolkit. Identifying the MUMPS operating system upon Toolkit's installation starts processes that create the appropriate environment. The ^%ZOSF global is built from an operating system-specific routine. By executing nodes of the ^%ZOSF global, implementation-specific functions that are not part of ANSI MUMPS are possible.

The %ZOSV routine contains code that enables use of the VIEW command and $VIEW function to get information from the operating system.

The Kernel allows processors running different operating systems to be linked. The ^%ZOSF global makes this possible, as well. ^%ZOSF is never translated and thus may retain processor-specific information.

^%ZOSF("OS") contains two pieces of information about the current operating system: the name and the internal entry number from the MUMPS OPERATING SYSTEM file (#.7). DISYS are set based on ^%ZOSF("OS"). If the ^%ZOSF global is defined, the VA FileMan init sends a task to the Manager's account to set the second piece of ^%ZOSF("OS"). The TaskMan option Check TaskMan's Environment [ZTMCHECK] displays the contents of ^%ZOSF("OS").

The Manager account is generally reserved for operating system-specific routines and globals. Part of Toolkit must also reside in this account to take care of certain input/output procedures.

The VAX/Alpha Performance Monitor (VPM) for Toolkit was developed and tested on Digital Equipment VAX systems using VAX DSM 6.2, VMS 5.5-2, as well as DSM for OpenVMS on Alpha.

DBA APPROVALS and DATABASE INTEGRATION AGREEMENTS (DBIAs)

1. INTEGRATION REFERENCE #295

NAME:	**DBIA295**	**ENTRY:**	**295**
CUSTODIAL PACKAGE:	**KERNEL**		**San Francisco**
SUBSCRIBING PACKAGE:	**TOOLKIT**		**San Francisco**
STATUS:	**ACTIVE**		
DURATION:	**Till Otherwise Agreed.**		

DESCRIPTION:

Integration Agreement Request between Toolkit (all versions) and Kernel (all versions).

Toolkit and Kernel agree that both packages shall distribute all routines and data for MUMPS operating system interfaces (e.g. ZOSF, ZOSV*).

Toolkit and Kernel also agree that the menus, [XUPROG], [XTMENU], and [XTCM MAIN], can be attached to the Kernel menu [EVE].

2. INTEGRATION REFERENCE #316

NAME: DBIA316-A **ENTRY:** 316
CUSTODIAL PACKAGE: VA FILEMAN San Francisco
SUBSCRIBING PACKAGE: TOOLKIT San Francisco
STATUS: ACTIVE
DURATION: Till Otherwise Agreed.

DESCRIPTION:

1. **When a new file is configured for use with MTLU, the variable-pointer "ENTRY" field is automatically updated in the LOCAL KEYWORD and LOCAL SHORTCUT files to reflect the new file. This must be handled via DIC/DIE calls with DIC/DIE being set to ^DD(file,.02,"V",. It is fully compatible with the interactive way of creating variable pointer type fields.**

2. **MTLU uses the string maintained in ^DD("KWIC"). There is currently no way of retrieving this information without directly referencing this node. As stated there is currently no way of extracting data stored in the node except by direct global hit.**

Amendment 5/11/94

Toolkit DBIA 316 has been amended to include the $ORDER of ^DD in line QU+5^XTLKEFOP. This code identifies the variable pointer prefix associated with the selected lookup file and was inadvertently omitted.

```
S XTLKY=Y,XTLKPF=+$O(^DD(8984.2,.02,"V","B",+Y,"")) G:'XTLKPF KL
S XTLKPF=$P(^DD(8984.2,.02,"V",XTLKPF,0),U,4),XTLKUT=1
```

GLOBAL REFERENCE:

^DD(D0,.02,'V',

^DD('KWIC')

3. INTEGRATION REFERENCE #833

NAME: DBIA316-B **ENTRY:** 833
CUSTODIAL PACKAGE: VA FILEMAN San Francisco
SUBSCRIBING PACKAGE: TOOLKIT San Francisco
STATUS: ACTIVE
DURATION: Till Otherwise Agreed.

DESCRIPTION:

3. **The look-up routine, XTLKDICL, is often executed recursively by FileMan. Under some conditions, it is not appropriate to proceed with the lookup and processing must pass back to DIC at the appropriate entry point. MTLU, therefore, needs support for the entry points ASK^DIC and RTN^DIC. Some of the variables that are used by the ASK^DIC and RTN^DIC calls are:**

Variables:	Used in:
DO(2	EN2+3,EN2+5
DIC	TS+1
DIC(0	XTLKDICL+3,EN1+2
DIE	XTLKDICL+3
DIPGM(0	XTLKDICL+3,XTLKDICL+5
DO	TS
DO(2	TS,TS+1,TS+2
X	XTLKDICL+4,EN2+1,EN2+3,EN2+5,TS+1,TS+4,TS+8,TS+9
Y	EN2+1,TS,TS+8,TS+9 Label References:
EN1	TS+9
EN2	XTLKDICL+5,TS+8

External References:

ASK^DIC	EN1+2
RTN^DIC	XTLKDICL+3,EN2+3,EN2+5

The calls to RTN^DIC and ASK^DIC are granted for the exclusive use of the Kernel's Toolkit package.

ROUTINE: DIC
COMPONENT: D0
VARIABLES:
COMPONENT: RTN
VARIABLES:

4. INTEGRATION REFERENCE #1062

NAME: 1062 ENTRY: 1062
CUSTODIAL PACKAGE: VA FILEMAN San Francisco
SUBSCRIBING PACKAGE: TOOLKIT San Francisco
STATUS: ACTIVE
DURATION: Till Otherwise Agreed.

DESCRIPTION:

Kernel Toolkit needs this agreement with FileMan to be able to use the variable D0 in DD definitions. Here are some examples of the use of variable D0.

15,99991	**LOOKUP1**	**; COMPUTED**
	MUMPS CODE:	**S X="`"_+^VA(15,D0,0)**
	ALGORITHM:	**S X="`"_+^VA(15,D0,0)**
	DESCRIPTION:	**This field is used to navigate to the file pointed to by RECORD1.**
	TECHNICAL DESCR:	**This field is used to navigate to the file pointed to by RECORD1.**
15,99992	**LOOKUP2**	**; COMPUTED**
	MUMPS CODE:	**S X="`"_+$P(^VA(15,D0,0),U,2)**
	ALGORITHM:	**S X="`"_+$P(^VA(15,D0,0),U,2)**
	DESCRIPTION:	**This field is used to navigate to the file pointed to by RECORD2.**
	TECHNICAL DESCR:	**This field is used to navigate to the file pointed to by RECORD2.**
15,99993	**LOOKUP3**	**; COMPUTED**
	MUMPS CODE:	**S X="`"_D0**
	ALGORITHM:	**S X="`"_D0**
	LAST EDITED:	**AUG 08, 1989**
	DESCRIPTION:	**This computed field provides navigational capability to any file that points to this file and has a DINUM relationship.**
	TECHNICAL DESCR:	**This computed field provides navigational capability to any file that points to this file and has a DINUM relationship.**

5. INTEGRATION REFERENCE #1091

NAME:	**DBIA316-C**	**ENTRY:**	**1091**
CUSTODIAL PACKAGE:	**VA FILEMAN**		**San Francisco**
SUBSCRIBING PACKAGE:	**TOOLKIT**		**San Francisco**
STATUS:	**ACTIVE**		
DURATION:	**Next Version**		

DESCRIPTION:

Multi Term Lookup (a component of TOOLKIT) requests the ability to read the "GL" node of ^DIC in order to retrieve a global root. This reference can be found in the routines XTLKEFOP, XTLKKWL, XTLKMGR, XTLKPRT, and in the MUMPS X-REF of file 8984.3 listed below:

CROSS-REFERENCE: 8984.3^AC^MUMPS

1)=
```
I $D(^XT(8984.3,DA,0)),$P(^(0),U,2)'="" S J
  L=$P(^(0),U,2),JL=$P(^DIC(JL,0,"GL"),U,2),^XT(8
                          984.3,"AC",JL,$E(X,1,30),DA)="" K JL
```

2)=
```
I $D(^XT(8984.3,DA,0)),$P(^(0),U,2)'="" S J
  L=$P(^(0),U,2),JL=$P(^DIC(JL,0,"GL"),U,2) K ^XT
                          (8984.3,"AC",JL,$E(X,1,30),DA),JL
```

Associates the synonym with the global root of the lookup file.

*** Amendment 1/23/95 ***

The above request should be modified to include both Multi-Term Lookup and the Duplicate Resolution modules of Toolkit. The "GL" node is referenced for the same purpose in file 15.1, field .01, "AGL" cross-reference.

6. INTEGRATION REFERENCE #1110

NAME:	**1110**	**ENTRY:**	**1110**
CUSTODIAL PACKAGE:	**VA FILEMAN**		**San Francisco**
SUBSCRIBING PACKAGE:	**TOOLKIT**		**San Francisco**
STATUS:	**ACTIVE**		
DURATION:	**Till Otherwise agreed**		

DESCRIPTION:

Kernel Toolkit needs this agreement with FileMan to be able to clean up some "IX" nodes in the data dictionary of the DUPLICATE RECORD (#15) file. The "IX" nodes which are killed during the post-init contain the names of the cross-references. which do not exist.

GLOBAL REFERENCE:

^DD(15,0,'IX','AMFI',15,999999901)

^DD(15,0,'IX','APOT',15,.04)

^DD(15,0,'IX','AZ1',15,.05)

^DD(15.01101,0,'IX','ARDY',15.01101,.02)

7. INTEGRATION REFERENCE #1111

NAME:	**1111**	**ENTRY:**	**1111**
CUSTODIAL PACKAGE:	**VA FILEMAN**		**San Francisco**
SUBSCRIBING PACKAGE:	**TOOLKIT**		**San Francisco**
STATUS:	**ACTIVE**		
DURATION:	**Till Otherwise agreed**		

DESCRIPTION:

Kernel Toolkit files have a number of fields whose screens, input transforms, and executable helps contain code that directly references ^DD.

GLOBAL REFERENCE:

^DD(15,.01,'V','B')

This node is used in the input transform and in the screen of the DUPLICATE RESOLUTION FILE 15.1. It is used in a variable pointer type field to restrict the user to only those files which have been set up for the merge.

^DD(FILE,FIELD)

These global references are used in displaying the fields that can be compared and assigned a matching value. The 0 node of the field is referenced in order to screen out "computed" and "multi-valued" fields from this display and comparison. One use of these references can be seen in the executable help of field .05, FIELD TO BE CHECKED.

^DD(FILE,FIELD,0)

These global references are used in displaying the fields that can be compared and assigned a matching value. The 0 node of the field is referenced in order to screen out "computed" and "multi-valued" fields from this display and comparison. One use of these references can be seen in the executable help of field .05, FIELD TO BE CHECKED.

8. INTEGRATION REFERENCE #1113

NAME: 1113 **ENTRY:** 1113
CUSTODIAL PACKAGE: KERNEL San Francisco
SUBSCRIBING PACKAGE: TOOLKIT San Francisco
STATUS: ACTIVE
DURATION: Till Otherwise agreed

DESCRIPTION:

Kernel Toolkit needs this agreement with Kernel to reference ^DIC(9.4.

GLOBAL REFERENCE:

^DIC(9.4,D0,20,D1,0)
3 NAME OF MERGE ROUTIN 0;3 Direct Global Read

^DIC(9.4,D0,20,D1,1)

^DIC(9.4,D0,0)
.01 NAME 0;1 Read w/Fileman

9. INTEGRATION REFERENCE #1124

NAME: References to PACKAGE FILE (9.4)
ENTRY: 1124
CUSTODIAL PACKAGE: KERNEL (parent) San Francisco
SUBSCRIBING PACKAGE: TOOLKIT San Francisco
STATUS: ACTIVE
DURATION: Till Otherwise agreed

DESCRIPTION:

^XTSUMBLD, %INDEX, and the XINDEX routines need to look at the Package file to find out what files are part of the package. For example,

```
>>>>>XTSUMBLD+14 (FIELD: PREFIX)
S X=$P(^DIC(9.4,+$P(Y(0),U,2),0),U,2) D NAME(X) G EXIT:'$D(XTRNAME)

>>>>>XINDX10+11 (FIELD: FILE)
F J=0:0 S J=$O(^DIC(9.4,DA,4,J)) Q:J'>0  I $D(^(J,0))
    SINDFN=+^(0),INDRN="|dd"| _INDFN,(INDF,INDL)=0 D INSERT

>>>>>XINDX11+22 (FIELD: PREFIX) NAMSP
S INDXN=$P(^DIC(9.4,DA,0),"^",2),C9=0,INDXN(C9)="," F A=0:0 S
    A=$O(^DIC(9.4,DA,"EX",A)) Q:A'>0  I $D(^(A,0))#2 S
    C9=C9+1,INDXN(C9)=$P(^(0),"^")
```

```
>>>>>ZINDX10+4 (FIELD: FILE)
F J=0:0 S J=$O(^DIC(9.4,DA,4,J)) Q:J'>0  I $D(^(J,0)) S
INDFN=+^(0),INDRN="|dd"| _INDFN,(INDF,INDL)=0 D INSERT

>>>>>ZINDX11+5 (FIELD: PREFIX) NAMSP
S INDXN=$P(^DIC(9.4,DA,0),"^",2),C9=0,INDXN(C9)="," F A=0:0 S
    A=$O(^DIC(9.4,DA,"EX",A)) Q:A'>0  I $D(^(A,0))#2 S
    C9=C9+1,INDXN(C9)=$P(^(0),"^")
```

GLOBAL REFERENCE:

^DIC(9.4,DA,0)
1 PREFIX 0;2 Direct Global Read

^DIC(9.4,DA,4)
6 *FILE 4;0 Direct Global Read

10. INTEGRATION REFERENCE #1125

NAME: Index and BUILD file
ENTRY: 1125
CUSTODIAL PACKAGE: KERNEL — San Francisco
SUBSCRIBING PACKAGE: TOOLKIT — San Francisco
STATUS: ACTIVE
DURATION: Till Otherwise agreed

DESCRIPTION:

Index reads the file list, option list, Function list, routine list to get the components of a build. The references are in XINDX10, XINDX11, XINDX51.

GLOBAL REFERENCE:

^XPD(9.6,D0,4
4 FILE Direct Global Read

^XPD(9.6,D0,'KRN',
6 BUILD COMPONENTS Direct Global Read

11. INTEGRATION REFERENCE #1126

NAME: Index and the DD global.
ENTRY: 1126
CUSTODIAL PACKAGE: VA FILEMAN — San Francisco
SUBSCRIBING PACKAGE: TOOLKIT — San Francisco
STATUS: ACTIVE
DURATION: Till Otherwise agreed

DESCRIPTION:

GLOBAL REFERENCE:

^DD(

The VA Cross-Referencer utility in Toolkit needs to reference several ^DD nodes in order to cross-reference a package. Several of the referenced DD nodes contain MUMPS code. They are inspected to find items such as global/variables names and label/external references. Some of the referenced nodes are the "LAYGO", "DEL" nodes. A specific example of a DD reference follows:

```
>>>>>%INDX10+25
S INDEL="" F G=0:0 S INDEL=$O(^DD(INDFN,INDF,"LAYGO",INDEL))
Q:INDEL=""
I $D(^(INDEL,0))#2 S INDC=INDF_"LAYGO"_INDEL_" ; LAYGO CHECK
CODE",INDX=^(0) D ADD
```

The DD references are found in routines %INDX10, %INDX11, %INDX53.

12. INTEGRATION REFERENCE #1129

NAME:	**DBIA1129-A**	**ENTRY:**	**1129**
CUSTODIAL PACKAGE:	**KERNEL (parent)**		**San Francisco**
SUBSCRIBING PACKAGE:	**TOOLKIT**		**San Francisco**
STATUS:	**ACTIVE**		
DURATION:	**Till Otherwise agreed**		

DESCRIPTION:

Reference to ^ZZSLOT. Toolkit requests read access to this node to maintain the number of active slots in its performance database.

```
.S XUCMSLOT=+$G(^ZZSLOT(XUCMND,"ACTIVE"))
```

GLOBAL REFERENCE:

^ZZSLOT(nodename,'ACTIVE')
 # active slots on this node.

13. INTEGRATION REFERENCE #1130

NAME: DBIA1129-B **ENTRY:** 1130
CUSTODIAL PACKAGE: KERNEL (parent) San Francisco
SUBSCRIBING PACKAGE: TOOLKIT San Francisco
STATUS: ACTIVE
DURATION: Till Otherwise agreed

DESCRIPTION:

References to ^%ZOSV*

ROUTINE: %ZOSV2

COMPONENT: DB
VARIABLES: XUCM() Output
Collect data on current database size.

COMPONENT: RTHSTOP
VARIABLES: Stops the current RTHIST session, prepares data for filing by moving to the %ZRTL global, purges ^RTH global in the MGR account, begins a new RTHIST session.

COMPONENT: $$TRNLNM
VARIABLES: Translates a VMS logical name.

COMPONENT: $$TI
VARIABLES: Returns MSM CPU tic interval.

COMPONENT: $$OS
VARIABLES: Return current operating system and version level.

COMPONENT: $$PRV
VARIABLES: Return current user priv's on VMS systems.

Internal Relations

RELATIONSHIP OF TOOLKIT WITH KERNEL AND VA FILEMAN

Toolkit requires both Kernel V. 7.1 and VA FileMan V. 20.0. Toolkit resides on the Kernel's Systems Manager Menu [EVE].

Any Multi-Term Look-Up option in the OPTION file (#19), which is a menu option, is able to run independently provided the user has the appropriate keys.

NAMESPACING

In production accounts Toolkit follows the namespacing conventions of DHCP primarily using a leading X. Within the X namespace XDR is the Duplicate Resolution Utilities, XUCM, and XUCS contain Capacity Management utilities, and XT has a set of tools supporting distinct tasks (e.g., XTLK is the namespace for the Multi-Term Look-Up utility). Toolkit also uses the Z namespace within the production account (e.g., ZIND).

NOTE: For absolute safety it is recommended that ZZ be used in local development.

Package-wide Variables

The Kernel Toolkit does not create any package-wide variables that have received SACC exemptions.

SACC Exemptions

The following list describes exemptions from the *Programming Standards and Conventions (SAC)* that currently pertain to Toolkit. The date the exemption was granted is shown in parentheses following the description.

- **The following globals are exempt from VA FileMan compatibility (8/10/89):**

  ^%ZRTL(3
  ^%ZRTL("RTH"

- **The Kernel routine ZTEDIT3 may SET or KILL the variable DUZ: (6/18/90).**

  NOTE: ZTEDIT3 is now a Toolkit Routine.

- **Kernel Toolkit may use the following Type A extensions to the 1990 MUMPS Language Standard:**

  Merge
  reverse $ORDER/2-arg $O
  $GET with two arguments
  $NAME
  SET $EXTRACT
  missing parameters in calling list
  set $X and $Y
  10K routine size

How to Generate On-Line Documentation

On-line documentation about Toolkit may be obtained in a number of ways as described in this chapter.

RETRIEVING ON-LINE HELP USING QUESTION MARKS

The use of question marks at the file and field level is described in the *VA FileMan Technical Manual*. The use of question marks within the menu system invoke help about options and menus. One question mark at the top-level menu prompt displays the items available on the menu. Two question marks shows the Common Menu available to all users as well as any secondary menu options for the current user. Locked options are displayed if the user holds the key. Three question marks displays descriptions of the options from the OPTION file (#19). Four question marks displays a help frame if one has been associated with this option in the OPTION file (#19). A question mark followed by the name of an option on the current menu displays a help frame if one has been named for that option in the OPTION file (#19).

PRINT OPTIONS FILE

The Kernel Print Option File option [XUPRINT] displays a list of namespaced options associated with VA FileMan and the Kernel. Other namespaced entries may also be retrieved from the following files:

PRINT TEMPLATE (#.4)

INPUT TEMPLATE (#.402)

SORT TEMPLATE (#.401)

SECURITY KEY (#19.1)

FUNCTION (#.5)

BULLETIN (#3.6)

HELP FRAME (#9.2)

LIST FILE ATTRIBUTES

The FileMan List File Attributes option [DILIST] allows the user to generate documentation pertaining to files and file structure. Utilization of this option via the "Standard" format yields the following Data Dictionary information for a specified file(s):

- **File name and description.**
- **Identifiers.**
- **Cross-references.**
- **Files pointed to by the file specified.**
- **Files which point to the file specified.**
- **Input templates.**
- **Print templates.**
- **Sort templates.**

In addition, the following applicable data is supplied for each field in the file: field name, number, title, global location, description, help prompt, cross-reference(s), input transform, date last edited, and notes.

Using the "Global Map" format of this option generates an output which lists all cross-references for the file selected, global location of each field in the file, input templates, print templates, and sort templates.

INQUIRE TO OPTION FILE

The Kernel Inquire option [XUINQUIRE] provides the following information about a specified option(s):

- **Option name.**
- **Menu text.**
- **Option description.**
- **Type of option.**
- **Lock (if any).**

In addition, all items on the menu are listed for each menu option.

To secure information about Multi-Term Look-Up options, the user must specify the name or namespace of the option(s) desired. The namespace associated with the Multi-Term Look-Up package is XTLK.

KERNEL HELP

The Kernel New Features Help option [XUVERSIONNEW-HELP] lists the help frames associated with the Kernel. Extensive information is available and the reader is encouraged to display or print this series of frames.

Checksum Values for Routines

This chapter contains the checksum values for the Toolkit's routines. These values reflect the checksum at the time of the package release. Subsequent changes (patches) to the routines change these values.

XDRCNT 7651887
XDRDADD 8133407
XDRDADJ 4509269
XDRDCOMP 4431965
XDRDFPD 7816795
XDRDLIST 7786460
XDRDMAIN 5700890
XDRDOC 19083
XDRDOC1 13351
XDRDOC2 19767
XDRDPDTI 2075925
XDRDPRGE 3959904
XDRDQUE 9275556
XDRDSCOR 1855732
XDRDSTAT 2676366
XDRDUP 3547600
XDREMSG 4302480
XDRERR 127648
XDRHLP 2681700
XDRMADD 6382715
XDRMAIN 7563507
XDRMAINI 14611797
XDRMPACK 2927651
XDRMRG 14311248
XDRMRG1 1874512
XDRMSG 1827956
XDRMVFY 1318075
XDRPREI 293004
XDRU1 1782236
XINDEX 7227772
XINDX1 6096231
XINDX10 12585180
XINDX11 7471101
XINDX2 5054188
XINDX3 3897455
XINDX4 4711071
XINDX5 6259999
XINDX51 9529173
XINDX52 2298647
XINDX53 4122188
XINDX6 10179476
XINDX7 7575886
XINDX8 6101428
XINDX9 4045898
XTBASE 2331979
XTCMFILN 4125344
XTEDTVXD 1542362
XTFC0 11055774
XTFC1 14547133
XTFCE 5859522
XTFCE1 6311273
XTFCR 5587602
XTFCR1 3692308
XTINEND 5215462
XTINI001 5950864
XTINI002 4393549
XTINI003 6019987
XTINI004 4028880
XTINI005 3990558
XTINI006 9416677
XTINI007 9086371
XTINI008 8419298
XTINI009 8019911
XTINI00A 9113926
XTINI00B 11110278
XTINI00C 11151493
XTINI00D 9306443
XTINI00E 8494510
XTINI00F 8489467
XTINI00G 7747693
XTINI00H 7094018
XTINI00I 7643278
XTINI00J 8405097
XTINI00K 7198108
XTINI00L 6651500
XTINI00M 8180768
XTINI00N 7132260
XTINI00O 2910462
XTINI00P 7694041
XTINI00Q 8160242
XTINI00R 9440499
XTINI00S 9016307

XTINI00T 9491800
XTINI00U 10283373
XTINI00V 8766708
XTINI00W 6509886
XTINI00X 8595714
XTINI00Y 7948819
XTINI00Z 1773424
XTINI010 7071607
XTINI011 5531060
XTINI012 7968888
XTINI013 11142565
XTINI014 2687098
XTINI015 8767076
XTINI016 3859202
XTINI017 7490625
XTINI018 2613367
XTINI019 4541347
XTINI01A 7350290
XTINI01B 3719011
XTINI01C 6289769
XTINI01D 1212716
XTINI01E 5998915
XTINI01F 5482770
XTINI01G 3469421
XTINI01H 1876516
XTINI01I 5948679
XTINI01J 5624949
XTINI01K 7118498
XTINI01L 5020375
XTINI01M 6515584
XTINI01N 7612374
XTINI01O 7804125
XTINI01P 7864176
XTINI01Q 7980433
XTINI01R 7872517
XTINI01S 7885668
XTINI01T 8156338
XTINI01U 5743708
XTINI01V 8379152
XTINI01W 7143097
XTINI01X 6494785
XTINI01Y 6468654
XTINI01Z 6344717
XTINI020 6053332
XTINI021 6154942
XTINI022 6263758
XTINI023 6988435
XTINI024 7095170
XTINI025 5225531
XTINI026 6836611
XTINI027 7165347
XTINI028 7867999
XTINI029 5902113
XTINI02A 7114778
XTINI02B 6562588
XTINI02C 5066519
XTINI02D 4746121
XTINI02E 3747162
XTINI02F 1763576
XTINIS 2134872
XTINIT 11072830
XTINIT1 5762600
XTINIT2 5232093
XTINIT3 16090016
XTINIT4 3357263
XTINIT5 1525744
XTINITY 15382450
XTINOK 2394003
XTKERM1 5596187
XTKERM2 7359658
XTKERM3 2782884
XTKERM4 5378382
XTKERMIT 2016322
XTLATSET 6413686
XTLKDICL 2562328
XTLKEFOP 12288261
XTLKKSCH 5117176
XTLKKWL 2673960
XTLKKWL1 8089076
XTLKKWL2 8570562
XTLKKWLD 830939
XTLKMGR 8218132
XTLKPRT 3890354
XTLKPST 561010
XTLKTICD 2688040
XTLKTOKN 3207127
XTLKWIC 2000831
XTRCMP 4749536
XTRGRPE 342530
XTRTHV 6157862
XTSPING 258974
XTSUMBLD 10426704
XTVCHG 2433675
XTVGC1 22093368
XTVGC1A 6994193
XTVGC2 20558103
XTVGC2A 17354751
XTVGC2A1 9865117

XTVNUM 7898211
XTVRC1 9444013
XTVRC1A..... 18908374
XTVRC1Z 573177
XTVRC2 18916359
XTVRCRES 5210571
XUCIN001 6331531
XUCIN002 7847815
XUCIN003 5328738
XUCIN004 5633578
XUCIN005 4016858
XUCIN006 6384285
XUCIN007 2842241
XUCIN008 4454633
XUCIN009 7054787
XUCIN00A..... 3478716
XUCIN00B..... 3162587
XUCIN00C..... 4414148
XUCIN00D..... 5426793
XUCIN00E..... 6950804
XUCIN00F 6487754
XUCIN00G..... 5536118
XUCIN00H 5595361
XUCIN00I 3636839
XUCIN00J 4985882
XUCIN00K..... 1326743
XUCIN00L 1575674
XUCIN00M 5562214
XUCIN00N 5268349
XUCIN00O..... 4317063
XUCIN00P 5661675
XUCIN00Q..... 7342486
XUCIN00R..... 8641639
XUCIN00S 5853502
XUCIN00T..... 5815554
XUCIN00U 6204905
XUCIN00V..... 4353627
XUCIN00W 5144510
XUCIN00X..... 5630258
XUCIN00Y..... 6331276
XUCIN00Z 7705187
XUCIN010 5146314
XUCIN011 2024106
XUCINIS..... 2173432
XUCINIT..... 10781726
XUCINIT1..... 5752814
XUCINIT2..... 5232654
XUCINIT3..... 16094813
XUCINIT4..... 3357826
XUCINIT5..... 1367458
XUCMBR1 5837802
XUCMBR2 10844732
XUCMBR3 9625086
XUCMBRTL 8754496
XUCMDSL 4295323
XUCMFGI..... 1467166
XUCMFIL 5382924
XUCMGRAF 1687213
XUCMNI2A..... 20928196
XUCMNIT..... 11960925
XUCMNIT1..... 7377867
XUCMNIT2..... 16835662
XUCMNIT3..... 5784566
XUCMNIT4..... 11052588
XUCMNIT5..... 4264655
XUCMNT3A..... 10767827
XUCMPA 7085998
XUCMPA1 7618346
XUCMPA2 6586755
XUCMPA2A..... 5655040
XUCMPA2B..... 9904709
XUCMPOST..... 1750081
XUCMPRE..... 2500182
XUCMTM..... 9551796
XUCMTM1..... 3008863
XUCMVPG..... 4016494
XUCMVPG1..... 5894133
XUCMVPI 5930227
XUCMVPM 4086669
XUCMVPM1 11280175
XUCMVPS 6211427
XUCMVPU 3071852
XUCPCLCT..... 3573145
XUCPFRMT..... 13051323
XUCPRAW..... 13134609
XUCS1E 5744464
XUCS1R 11414218
XUCS1RA..... 11144520
XUCS1RB..... 11341248
XUCS1RBA..... 6176633
XUCS2E 5426963
XUCS2R 7996923
XUCS2RA..... 6913779
XUCS2RB..... 8006304
XUCS2RBA..... 4179178
XUCS4E 1556880
XUCS4R 11653758
XUCS4RB..... 9766381

XUCS5E 1037983
XUCS5EA..... 5223554
XUCS6E 1362981
XUCS6R 6132484
XUCS8E 2709944
XUCS8R 12237493
XUCS8RB..... 10517473
XUCS8RG 5086949
XUCS8RGA..... 4599537
XUCSCDE..... 3642152
XUCSCDG 6572594
XUCSCDGA..... 4312009
XUCSCDR..... 9885385
XUCSCDRB 8466812
XUCSI001 6807214
XUCSI002 8095362
XUCSI003 7090131
XUCSI004 6290195
XUCSI005 3935311
XUCSI006 4874125
XUCSI007 6391003
XUCSI008 6245860
XUCSI009 6641536
XUCSI00A..... 7072648
XUCSI00B..... 6183955
XUCSI00C..... 5840656
XUCSI00D 5871453
XUCSI00E..... 2444351
XUCSI00F..... 850326
XUCSI00G 5117155
XUCSI00H 7689858
XUCSI00I..... 6887131
XUCSI00J 6573491
XUCSI00K 6171637
XUCSI00L..... 4221793
XUCSINI1..... 5671874
XUCSINI2..... 5232622
XUCSINI3..... 16094695
XUCSINI4..... 3357794
XUCSINI5..... 628012
XUCSINIS 2216765
XUCSINIT 10890951
XUCSLOAD 6778573
XUCSPRG..... 4520058
XUCSRV 5969322
XUCSTM..... 6457738
XUCSTME 13223145
XUCSUTL..... 2837344
XUCSUTL2..... 6021611
XUCSUTL3..... 11276783
XURTL 7949393
XURTL1 7623474
XURTL2 5911591
XURTL3 9463174
XURTL4 8083788
XURTLC 3647421
XURTLK 5463315
ZINDEX..... 7934389
ZINDX1 5876099
ZINDX10 11716270
ZINDX11 6160284
ZINDX2 4603647
ZINDX3 3896672
ZINDX4 4546127
ZINDX5 6415684
ZINDX51 8782796
ZINDX52 2299766
ZINDX53 4122137
ZINDX6 12035964
ZINDX8 6760814
ZINDX9 4986099
ZINDXH 1579327
ZTEDIT 11385452
ZTEDIT1 9783719
ZTEDIT2 12580728
ZTEDIT3 9890321
ZTEDIT4 4936626
ZTGS 1511640
ZTP1 7893577
ZTPP..... 7019346
ZTRDEL 959784
ZTRTHV..... 6018658

Security and Keys

The security keys distributed with Toolkit to protect the use of options are described below:

XDR	**This key allows access to the Duplicate Resolution options.**
XDRMGR	**This key allows a user access to the Duplicate Resolution Manager utilities. It should only be given to the people responsible for management of the various Duplicate Resolution packages (e.g., Patient Registration).**
XTLKZMGR	**This is a manager's security key used to lock the set and kill options of the LOCAL LOOKUP file (#8984.4).**
XTLKZUSER	**This security key may optionally be used to lock the XTLKUSER2 menu.**
XUPROG	**Assign this lock to all users allowed to go into programmer options from the Menu system.**
XUPROGMODE	**This key locks out "Global List" and "Programmer Mode".**

GLOBAL NAME	FILE #	FILE NAME *
DIZ	8980	KERMIT HOLDING
XTV	8991	XTV ROUTINE CHANGES
	8991.19	XTV VERIFICATION PACKAGE
	8991.2	XTV GLOBAL CHANGES
VA	15	DUPLICATE RECORD
	15.1	DUPLICATE RESOLUTION
%ZRTL	3.091	RESPONSE TIME
	3.092	RT DATE_UCI,VOL
	3.094	RT RAWDATA
^XT	8984.1	LOCAL KEYWORD
	8984.2	LOCAL SHORTCUT
	8984.3	LOCAL SYNONYM
	8984.4	LOCAL LOOKUP
^XUCM	8986.095	CM SITE PARAMETERS
	8986.098	CM BERNSTEIN DATA
	8986.3	CM SITE NODENAMES
	8986.35	CM SITE DISKDRIVES
	8986.4	CM METRICS
	8986.5	CM DISK DRIVE RAW DATA
	8986.51	CM NODENAME RAW DATA
	8986.6	CM DAILY STATISTICS
^XUCS	8987.1	MSM RTHIST SITE
	8987.2	MSM RTHIST REPORT DATA

Global Translation

This chapter contains recommendations for journaling and translating Toolkit globals. (Translation is called "Impliciting" when running M/SQL.) Also, globals that should exist independently on each CPU are shown. The *Kernel Toolkit Installation Guide V. 7.3* has additional information regarding these issues.

NOTE: It is recommended, but not necessary, that the ^XT global be journalled.

Sites using MSM should consult the *486 Cookbook and MSM System Managers Guide* for instructions and recommendations regarding journaling, translation, and replication; the information here may not apply.

TRANSLATION *(or IMPLICIT for M/SQL)*				
Highly Recommended:	**^DIZ**	**^VA**	**^XTV**	
	^XUCM	**^XT**	**^XUCP**	**^XUCS**
Recommended:	**^%ZRTL**			

SEPARATE COPIES ON EACH CPU		
Permissible: (acceptable)	**^%ZRTL**	**(to have separate reports per CPU.)**

Journaling is recommended for the ^XT global.

Mapping Routines

Routine mapping is at the discretion of the systems manager. The RTHIST routines provide a method for each site to determine the extent to which certain routines are utilized.

The following list is provided only as a recommendation. See the *Kernel Technical Manual* and the *VA FileMan Technical Manual* for recommendations for mapping routines in those packages.

The following routine would be mapped in the Manager account:

%ZOSV (To avoid potential problems, do not map %ZOSV if you are running a version of VAX DSM less than V6.)

Glossary

ACCESS CODE — **A code that, along with the verify code, allows the computer to identify you as a user authorized to gain access to the computer. Your code is greater than six and less than twenty characters long; can be numeric, alphabetic, or a combination of both; and is usually assigned by a site manager or application coordinator . It is used by the Kernel's Sign-on/Security system to identify the user (see Verify Code).**

ALERTS — **Brief on-line notices that are issued to users as they complete a cycle through the menu system. Alerts are designed to provide interactive notification of pending computing activities, such as the need to reorder supplies or review a patient's clinical test results. Along with the alert message is an indication that the View Alerts common option should be chosen to take further action.**

ANS MUMPS — **The MUMPS programming language is a standard, that is an American National Standard (ANS). MUMPS stands for Massachusetts Utility Multi-programming System.**

ANSI — **American National Standards Institute**

APPLICATION PACKAGE — **In DHCP, software and documentation that support the automation of a service, such as Laboratory or Pharmacy within VA medical centers (see Package). The Kernel is like an operating system relative to other DHCP applications.**

APPLICATION PROGRAMMER — **The person who writes code for application packages. The Kernel provides tools to facilitate package development.**

APPLICATION PROGRAMMING INTERFACE (API) — **Programmer calls provided by the Kernel for use by application programmers. APIs allow programmers to carry out standard computing activities without needing to duplicate Kernel utilities in their own packages. APIs also further DBA goals of system integration by channeling activities, such as adding new users, through a limited number of callable entry points.**

ARRAY — **An arrangement of elements in one or more dimensions. A MUMPS array is a set of nodes referenced by subscripts which share the same variable name.**

AUTO-MENU	**An indication to Menu Manager that the current user's menu items should be displayed automatically. When auto-menu is not in effect, the user must enter a question mark at the menu's select prompt to see the list of menu items.**
BULLETINS	**Electronic mail messages that are automatically delivered by MailMan under certain conditions. For example, a bulletin can be set up to fire when database changes occur, such as adding a record to the file of users. Bulletins are fired by bulletin-type cross-references.**
CALLABLE ENTRY POINT	**An authorized programmer call that may be used in any DHCP application package. The DBA maintains the list of DBIC-approved entry points.**
CAPACITY MANAGEMENT	**The process of assessing a system's capacity and evaluating its efficiency relative to workload in an attempt to optimize system performance. The Kernel Toolkit provides several utilities which aid in the short and long term decision process of hardware and application code optimization.** **New Capacity Management Utilities have been created to utilize VMS, MUMPS and the latest VA Kernel Utilities. These utilities sample running systems at regular intervals and store a key subset of systems metrics related to configuration, database activity, response time, CPU, memory, and I/O utilization.**
COMMON MENU	**Options that are available to all users. Entering two question marks at the menu's select prompt displays any secondary menu options available to the signed-on user along with the common options available to all users.**
COMPILED MENU SYSTEM (^XUTL GLOBAL)	**Job-specific information that is kept on each CPU so that it is readily available during the user's session. It is stored in the ^XUTL global, which is maintained by the menu system to hold commonly referenced information. The user's place within the menu trees is stored, for example, to enable navigation via menu jumping.**
CPT	**Current Procedural Terminology**
CROSS REFERENCE	**An indexing method whereby files can include pre-sorted lists of entries as part of the stored database. Cross-references facilitate look-up and reporting.**

DATA — A representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means. The information you enter for the computer to store and retrieve. Characters that are stored in the computer system as the values of local or global variables. VA FileMan fields hold data values for file entries.

DATA ATTRIBUTE — A characteristic of a unit of data such as length, value, or method of representation. VA FileMan field definitions specify data attributes.

DATA DICTIONARY — The Data Dictionary is a global containing a description of what kind of data is stored in the global corresponding to a particular file. The data is used internally by FileMan for interpreting and processing files.

A Data Dictionary (DD) contains the definitions of a file's elements (fields or data attributes), relationships to other files, and structure or design. Users generally review the definitions of a file's elements or data attributes; programmers review the definitions of a file's internal structure.

DATABASE — A set of data, consisting of at least one file, that is sufficient for a given purpose. The Kernel database is composed of a number of VA FileMan files.

DBA — Database Administrator. In DHCP, the person who monitors namespacing conventions and other procedures that enable various DHCP packages to coexist within an integrated database system.

DBIA — Database Integration Agreement, a formal understanding between two or more DHCP packages which describes how data is shared or how packages interact. The DBA maintains a list of DBIAs between package developers allowing the use of internal entry points or other package-specific features that are not available to the general programming public.

DBIC — Database Integration Committee. Within the purview of the DBA, the committee maintains a list of DBIC-approved callable entry points and publishes the list on FORUM for reference by application programmers and verifiers.

DEVICE — A peripheral connected to the host computer, such as a printer, terminal, disk drive, modem, and other types of hardware and equipment associated with a computer. The host files of underlying operating systems may be treated like devices in that they may be written to (e.g., for spooling).

DEVICE HANDLER — The Kernel module that provides a mechanism for accessing peripherals and using them in controlled ways (e.g., user access to printers or other output devices).

DHCP — The Decentralized Hospital Computer Program of the Veterans Health Administration (VHA), Department of Veterans Affairs (VA). DHCP software, developed by VA, is used to support clinical and administrative functions at VA Medical Centers nationwide. It is written in MUMPS and, via the Kernel, runs on all major MUMPS implementations regardless of vendor. DHCP is composed of packages which undergo a verification process to ensure conformity with namespacing and other DHCP standards and conventions.

DICTIONARY — A database of specifications of data and information processing resources. VA FileMan's database of Data Dictionaries is stored in the FILE of files (#1).

DIFROM — VA FileMan utility that gathers all package components and changes them into routines (namespaceI* routines) so that they can be exported and installed in another VA FileMan environment.

DIRECT MODE UTILITY — A programmer call that is made when working in direct programmer mode. A direct mode utility is entered at the MUMPS prompt (e.g., >D ^XUP). Calls that are documented as direct mode utilities *cannot* be used in application package code.

DOUBLE QUOTE (") — A symbol used in front of a Common option's menu text or synonym to select it from the Common menu. For example, the five character string "TBOX" selects the User's Toolbox Common option.

DR STRING	**The set of characters used to define the variable DR when calling VA FileMan. Since a series of parameters may be included within quotes as a literal string, the variable's definition is often called the DR string. To define the fields within an edit sequence, for example, the programmer may specify the fields using a DR string rather than an input template.**
DUPLICATE RESOLUTION UTILITIES	**The Merge Shell was developed by the Indian Health Service (IHS) to support their Multi-Facility Integration project. Duplicate Resolution Utilities provide the functionality of combining duplicate records based on conditions established in customized applications.**
DUZ	**A local variable holding the user number that identifies the signed-on user.**
DUZ(0)	**A local variable that holds the File Manager Access Code of the signed-on user.**
ELECTRONIC SIGNATURE CODE	**A secret password that some users may need to establish in order to sign documents via the computer.**
ENTRY	**A VA FileMan record. It is uniquely identified by an internal entry number (the .001 field) in a file.**
ERROR TRAP	**A mechanism to capture system errors and record facts about the computing context such as the local symbol table, last global reference, and routine in use. Operating systems provide tools such as the %ER utility. The Kernel provides a generic error trapping mechanism with use of the ^%ZTER global and ^XTER* routines. Errors can be trapped and, when possible, the user is returned to the menu system.**
EXTRINSIC FUNCTION	**An extrinsic function is an expression that accepts parameters as input and returns a value as output that can be directly assigned.**
FIELD	**In a record, a specified area used for the value of a data attribute. The data specifications of each VA FileMan field are documented in the file's Data Dictionary. A field is similar to blanks on forms. It is preceded by words that tell you what information goes in that particular field. The blank, marked by the cursor on your terminal screen, is where you enter the information.**

FILE	**A set of related records treated as a unit. VA FileMan files maintain a count of the number of entries or records.**
FILE MANAGER (VA FILEMAN)	**The DHCP's Database Management System (DBMS). The central component of the Kernel that defines the way standard DHCP files are structured and manipulated.**
FORCED QUEUING	**A device attribute indicating that the device can only accept queued tasks. If a job is sent for foreground processing, the device rejects it and prompts the user to queue the task instead.**
FORM	**A screen-oriented display (see ScreenMan).**
FORUM	**The central E-mail system within DHCP. It is used by developers to communicate at a national level about programming and other issues. FORUM is located at the Washington, DC ISC (162-2).**
GLOBAL VARIABLE	**A variable that is stored on disk (MUMPS usage).**
GO-HOME JUMP	**A menu jump that returns the user to the Primary menu presented at sign-on. It is specified by entering two up-arrows (^^) at the menu's select prompt. It resembles the rubber band jump but without an option specification after the up-arrows.**
HELP FRAMES	**Entries in the HELP FRAME file (#9.2) that may be distributed with application packages to provide on-line documentation. Frames may be linked with other related frames to form a nested structure.**
HELP PROCESSOR	**A Kernel module that provides a system for creating and displaying on-line documentation. It is integrated within the menu system so that help frames associated with options can be displayed with a standard query at the menu's select prompt.**
HOOK OR LINK	**Non-specific terms referring to ways in which files may be related (via pointer links) or can be accessed (via hooks).**
HOST FILE SERVER (HFS)	**A procedure available on layered systems whereby a file on the host system can be identified to receive output. It is implemented by the Device Handler's Host File Server (HFS) device type.**

HUNT GROUP	**An attribute of an entry in the DEVICE file (#3.5) that allows several devices to be used interchangeably; useful for sending network mail or printing reports. If the first hunt group member is busy, another member may stand in as a substitute.**
ICD	**International Classification of Diseases**
INDEX (%INDEX)	**A Kernel utility used to verify routines and other MUMPS code associated with a package. Checking is done according to current ANSI MUMPS standards and DHCP programming standards (see SAC). This tool can be invoked through an option or from direct mode (>D ^%INDEX).**
INIT	**Initialization of an application package. INIT* routines are built by VA FileMan's DIFROM and, when run, recreate a set of files and other package components.**
INTERNAL ENTRY NUMBER (IEN)	**The number used to identify an entry within a file. Every record has a unique internal entry number.**
IRM	**Information Resource Management. A service at VA medical centers responsible for computer management and system security.**
JUMP START	**A logon procedure whereby the user enters the "access code;verify code;option" to go immediately to the target option, indicated by its menu text or synonym. The jump syntax can be used to reach an option within the menu trees by entering "access;verify;^option".**
KERMIT	**A standard file transfer protocol. It is supported by the Kernel and can be set up as an alternate editor.**
KERNEL	**A set of DHCP MUMPS software routines that function as an intermediary between the host operating system and the DHCP application packages enabling packages to coexist in a standard OS-independent computing environment. The Kernel provides a standard and consistent user and programmer interface between application packages and the underlying MUMPS implementations.**
KEYWORD	**A word or phrase used to call up several codes from the reference files in the LOCAL LOOKUP file (#8984.4). One specific code may be called up by several different keywords.**

LOCAL LOOKUP FILE	**The file into which sites enter selected reference files to be used in the look-up process.**
MAIL MESSAGE	**An entry in the MESSAGE file (#3.9). The DHCP electronic mail system (MailMan) supports local and remote networking of messages.**
MAILMAN	**The Kernel module that provides a mechanism for handling electronic communication, whether it is user-oriented mail messages, automatic firing of bulletins, or initiation of server-handled data transmissions.**
MANAGER ACCOUNT	**A UCI that can be referenced by non-manager accounts such as production accounts. Like a library, the MGR UCI holds percent routines and globals (e.g., ^%ZOSF) for shared use by other UCIs.**
MENU	**A list of choices for computing activity. A menu is a type of option designed to identify a series of items (other options) for presentation to the user for selection. When displayed, menu-type options are preceded by the word "Select" and followed by the word "option" as in Select Menu Management option: (the menu's select prompt).**
MENU CYCLE	**The process of first visiting a menu option by picking it from a menu's list of choices and then returning to the menu's select prompt. Menu Manager keeps track of information, such as the user's place in the menu trees, according to the completion of a cycle through the menu system.**
MENU MANAGER	**The Kernel module that controls the presentation of user activities such as menu choices or options. Information about each user's menu choices is stored in the Compiled Menu System, the ^XUTL global, for easy and efficient access.**
MENU SYSTEM	**The overall Menu Manager logic as it functions within the Kernel framework.**
MENU TEMPLATE	**An association of options as pathway specifications to reach one or more final destination options. The final options must be executable activities and not merely menus for the template to function. Any user may define user-specific menu templates via the corresponding Common option.**

MENU TEXT	**The descriptive words that appear when a list of option choices is displayed; specifically, the Menu Text field of the OPTION file (#19). For example, User's Toolbox is the menu text of the XUSERTOOLS option. The option's synonym is TBOX.**
MENU TREES	**The menu system's hierarchical tree-like structures that can be traversed or navigated, like pathways, to give users easy access to various options.**
MULTI-TERM LOOK-UP (MTLU)	**Multi-Term Look-Up (MTLU) is an adaptation of a tool developed by the Indian Health Service (IHS) which was made generic by the Albany ISC. Multi-Term Look-Up provides a method of enhancing the look-up capabilities of associated VA FileMan files.**
MULTIPLE	**A multiple-valued field; a subfile. In many respects, a multiple is structured like a file.**
MUMPS (ANSI STANDARD)	**A programming language recognized by the American National Standards Institute (ANSI). The acronym MUMPS stands for Massachusetts General Hospital Utility Multi-programming System.**
NAMESPACING	**A convention for naming DHCP package elements. The DBA assigns unique character strings for package developers to use in naming routines, options, and other package elements so that packages may coexist. The DBA also assigns a separate range of file numbers to each package.**
NODE	**In a tree structure, a point at which subordinate items of data originate. A MUMPS array element is characterized by a name and a unique subscript. Thus the terms: node, array element, and subscripted variable are synonymous. In a global array, each node might have specific fields or "pieces" reserved for data attributes such as name.**
OPERATING SYSTEM INDEPENDENCE (OS-INDEPENDENT)	**A key goal of DHCP. An insulation from specific features of the underlying operating system that allows application packages to run in different OS environments. The Kernel provides the interface mainly with use of the ^%ZOSF global.**

OPTION	**An entry in the OPTION file (#19). As an item on a menu, an option provides an opportunity for users to select it, thereby invoking the associated computing activity. Options may also be scheduled to run in the background, non-interactively, by TaskMan.**
OPTION NAME	**The Name field in the OPTION file (#19) (e.g., XUMAINT for the option that has the menu text "Menu Management"). Options are namespaced according to DHCP conventions monitored by the DBA.**
PAC	**Programmer Access Code. An optional user attribute that may function as a second level password into programmer mode.**
PACKAGE	**The set of programs, files, documentation, help prompts, and installation procedures required for a given software application. A DHCP software environment composed of elements specified via the Kernel's PACKAGE file (#9.4). Elements include files and associated templates, namespaced routines, and namespaced file entries from the OPTION (#19), SECURITY KEY (#19.1), HELP FRAME (#9.2), BULLETIN (#3.6), and FUNCTION (#.5) files. Packages are transported using VA FileMan's DIFROM routine that creates initialization routines to bundle the files and records for export. Installing a package involves the running of the installation routines that create the required software environment. Verified packages include documentation. As public domain software, verified packages may be requested through the Freedom of Information Act (FOIA).**
PHANTOM JUMP	**Menu jumping in the background. Used by the menu system to check menu pathway restrictions.**
POINTER	**A relationship between two VA FileMan files that makes navigation possible via the pointer (forward or backward).**
PRIMARY MENUS	**The list of options presented at sign-on. Each user must have a primary menu in order to sign-on and reach Menu Manager. Users are given primary menus by IRM. This menu should include most of the computing activities the user needs.**
PRODUCTION ACCOUNT	**The UCI where users log on and carry out their work, as opposed to the manager, or library, account.**

PROGRAMMER ACCESS — The ability to use DHCP features reserved for programmers. Having the programmer's at-sign, when DUZ(0)=@, enables programmer access.

PROMPT — The computer interacts with the user by issuing questions called *prompts*, to which the user issues a response.

PROTOCOL — An entry in the PROTOCOL file (#101). Used by the Order Entry/Results Reporting (OE/RR) package to support the ordering of medical tests and other activities. The Kernel includes several protocol-type options for enhanced menu displays within the OE/RR package.

QUEUING — Requesting that a job be processed in the background rather than in the foreground within the current session. Jobs are processed sequentially (first-in, first-out). The Kernel's Task Manager handles the queuing of tasks.

QUEUING REQUIRED — An option attribute that specifies that the option must be processed by TaskMan (the option can only be queued). The option may be invoked and the job prepared for processing, but the output can only be generated during the specified time periods.

RECORD — A set of related data treated as a unit. An entry in a VA FileMan file constitutes a record. A collection of data items that refer to a specific entity (e.g., in a name-address-phone number file, each record would contain a collection of data relating to one person).

RESOURCE — A method that enables sequential processing of tasks. The processing is accomplished with a RES device type designed by the application programmer and implemented by IRM. The process is controlled via the RESOURCE file (#3.54).

RETURN — On the computer keyboard, the key located where the carriage return is on an electric typewriter. It is used in DHCP to terminate "reads" and is symbolized by <RET>.

ROUTINE — A program or a sequence of instructions called by a program, that may have some general or frequent use. MUMPS routines are groups of program lines which are saved, loaded, and called as a single unit via a specific name.

RUBBER BAND JUMP	**A menu jump used to go out to an option and then return, in a bouncing motion. The syntax of the jump is two up-arrows followed by an option's menu text or synonym (e.g., ^^Print Option File). If the two up-arrows are not followed by an option specification, the user is returned to the primary menu (see Go-home Jump).**
SAC	**Standards and Conventions. Through a process of verification, DHCP packages are reviewed with respect to SAC guidelines as set forth by the Standards and Conventions Committee (SACC). Package documentation is similarly reviewed in terms of standards set by the Documentation Standards and Conventions Committee (DSCC).**
SACC	**DHCP's Standards and Conventions Committee. This Committee is responsible for maintaining the document called the SAC.**
SCHEDULING OPTIONS	**This is a technique of requesting that TaskMan run an option at a given time, perhaps with a given rescheduling frequency, such as once per week.**
SCREENMAN FORMS	**A screen-oriented display of fields, for editing or simply for reading. VA FileMan's Screen Manager is used to create forms that are stored in the FORM file (#.403) and exported with a package. Forms are composed of blocks [stored in the BLOCK file (#.404)] and can be regular, full screen pages or smaller, pop-up pages for multiples.**
SECONDARY MENUS	**Options assigned to individual users to tailor their menu choices. If a user needs a few options in addition to those available on the Primary menu, the options can be assigned as secondary options. To facilitate menu jumping, secondary menus should be specific activities, not elaborate and deep menu trees.**
SECURITY KEY	**The purpose of Security Keys is to set a layer of protection on the range of computing capabilities available with a particular software package. The availability of options is based on the level of system access granted to each user.**
SERVER	**An entry in the OPTION file (#19). An automated mail protocol that is activated by sending a message to the server with the "S.server" syntax. A server's activity is specified in the OPTION file (#19) and can be the running of a routine or the placement of data into a file.**

SHORTCUT — A word used to call up one specific code from the reference files in the LOCAL LOOKUP file (#8984.4).

SIGN-ON/SECURITY — The Kernel module that regulates access to the menu system. It performs a number of checks to determine whether access can be permitted at a particular time. A log of sign-ons is maintained.

SITE MANAGER/ IRM CHIEF — At each site, the individual who is responsible for managing computer systems, installing and maintaining new modules, and serving as liaison to the ISCs.

SPECIAL QUEUING — An option attribute indicating that TaskMan should automatically run the option whenever the system reboots.

SPOOLER — An entry in the DEVICE file (#3.5). It uses the associated operating system's spool facility, whether it is a global, device, or host file. The Kernel manages spooling so that the underlying OS mechanism is transparent. In any environment, the same method can be used to send output to the spooler. The Kernel subsequently transfers the text to the ^XMBS global for subsequent despooling (printing).

Spooling (under any system) provides an intermediate storage location for files (or program output) for printing at a later time.

SUBSCRIPT — A symbol that is associated with the name of a set to identify a particular subset or element. In MUMPS, a numeric or string value that: Is enclosed in parentheses, is appended to the name of a local or global variable, and identifies a specific node within an array.

SYNONYM — A field in the OPTION file (#19). Options may be selected by their menu text or synonym (see Menu Text).

In the case of Multi-Term Look-Up (MTLU), it is a word used to expand the call-up capability of existing terms in the LOCAL LOOKUP file (#8984.4).

TASKMAN — The Kernel module that schedules and processes background tasks (also called Task Manager).

TEMPLATES — **In VA FileMan, a means of storing report formats, data entry formats, and sorted entry sequences. A template is a permanent place to store selected fields for use at a later time. Edit sequences are stored in the INPUT TEMPLATE file (#.402), print specifications are stored in the PRINT TEMPLATE file (#.4), and search or sort specifications are stored in the SORT TEMPLATE file (#.401).**

TIMED-READ — **The amount of time the Kernel waits for a user response to an interactive READ command before starting to halt the process (times out).**

TOOLKIT — **Toolkit is a robust set of tools developed to aid the Decentralized Hospital Computer Program (DHCP) development community, and Information Resources Management (IRM), in writing, testing, and analysis of code. It is a set of generic tools that are used by developers, documenters, verifiers, and packages to support distinct tasks.**

Toolkit provides utilities for the management and definition of development projects. Many of these utilities have been used by the San Francisco Information Systems Center (ISC) for internal management and have proven valuable. Toolkit also includes tools provided by other ISCs based on their proven utility.

TREE STRUCTURE — **A term sometimes used to describe the structure of a MUMPS array. This has the same structure as a family tree, with the root at the top, and ancestor nodes arranged below, according to their depth of subscripting. All nodes with one subscript are at the first level, all nodes with two subscripts at the second level, and so on.**

TRIGGER — **A type of VA FileMan cross-reference. Often used to update values in the database given certain conditions (as specified in the trigger logic). For example, whenever an entry is made in a file, a trigger could automatically enter the current date into another field holding the creation date.**

TYPE-AHEAD — **A buffer used to store characters that are entered before the corresponding prompt appears. Type-ahead is a shortcut for experienced users who can anticipate an expected sequence of prompts.**

UCI
User Class Identification, a computing area. The MGR UCI is typically the Manager's account, while VAH or ROU may be Production accounts.

UP-ARROW JUMP
In the menu system, entering an up-arrow (^) followed by an option name accomplishes a jump to the target option without needing to take the usual steps through the menu pathway.

USER ACCESS
This term is used to refer to a limited level of access to a computer system which is sufficient for using/operating a package, but does not allow programming, modification to data dictionaries, or other operations that require programmer access. Any option, for example, can be locked with the key XUPROGMODE, which means that invoking that option requires programmer access.

The user's access level determines the degree of computer use and the types of computer programs available. The Systems Manager assigns the user an access level.

USER INTERFACE
The way the package is presented to the user such as issuing of prompts, help messages, menu choices, etc. A standard user interface can be achieved by using VA FileMan for data manipulation, the menu system to provide option choices, and VA FileMan's Reader, the ^DIR utility, to present interactive dialogue.

VA FILEMAN
A set of programs used to enter, maintain, access, and manipulate a database management system consisting of files. A package of on-line computer routines written in the MUMPS language which can be used as a stand-alone database system or as a set of application utilities. In either form, such routines can be used to define, enter, edit, and retrieve information from a set of computer stored files.

VARIABLE

A character, or group of characters, that refer to a value. MUMPS recognizes three types of variables:

1. **local variables**
2. **global variables**
3. **special variables**

Local variables exist in a partition of main memory and disappear at sign-off. A global variable is stored on disk, potentially available to any user. Global variables usually exist as parts of global arrays. The term "global" may refer either to a global variable or a global array. A special variable is defined by systems operations (e.g., $TEST).

VENDOR INDEPENDENCE

A goal of DHCP: To develop a system that does not assume the existence of a particular hardware/software platform supplied by a particular vendor. (See Operating System Independence.)

VERIFICATION

A process of DHCP package review carried out by technical staff not directly involved in the development of the package. Software and associated documentation are reviewed in terms of the *Programming Standards and Conventions (SAC)*.

VERIFY CODE

The Kernel's Sign-on/Security system uses the verify code to validate the user's identity. This is an additional security precaution used in conjunction with the Access Code. Like the Access Code, it is also 6 to 20 characters in length. If entered incorrectly, it does not allow the user to access the computer. To protect the user, both codes are invisible on the terminal screen.

Z EDITOR (^%Z)

A Kernel tool used to edit routines or globals. It can be invoked with an option, or from direct mode after loading a routine with >X ^%Z.

ZOSF GLOBAL (^%ZOSF)

The MUMPS OPERATING SYSTEM file (#.7) is a Manager account global distributed with the Kernel to provide an interface between DHCP application packages and the underlying operating system. This global is built during Kernel installation when running the manager setup routine (ZTMGRSET). The nodes of the global are filled-in with operating system-specific code to enable interaction with the operating system. Nodes in the ^%ZOSF global may be referenced by programmers so that separate versions of the package need not be written for each operating system (see Operating System Independence).

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

V

X

Z

Department of Veterans Affairs
Decentralized Hospital Computer Program

KERNEL TOOLKIT
TECHNICAL MANUAL

Version 7.3

April 1995

Information Systems Center
San Francisco, California

Preface

The purpose of this manual is to provide information about the structure of the set of software utilities known as the Kernel Toolkit (also referred to as "Toolkit"). This manual consists of technical material specifically intended for DHCP systems managers and developers.

Table of Contents